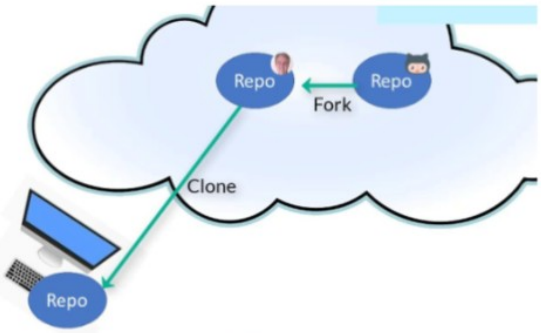
CLONE VS FORK

- Forking and cloning are, in many ways, extremely similar actions.
- Both effectively copy the entire source code of a repository.
- What sets them apart is the intention how you want to use.

- **Clone** → This will create a local clone of the repository (in your computer.) After this, you can make whatever changes you like in your IDE and pull/push.

- **Fork** → basically is taking someone else's repository under their Github account and making a copy of this repository under your github account. (NOTE: After fork, repository is still in Github, not your computer)
  - Essentially you can then **clone your forked repository** to your local.



Fork a Repository on GitHub (Manual Step)

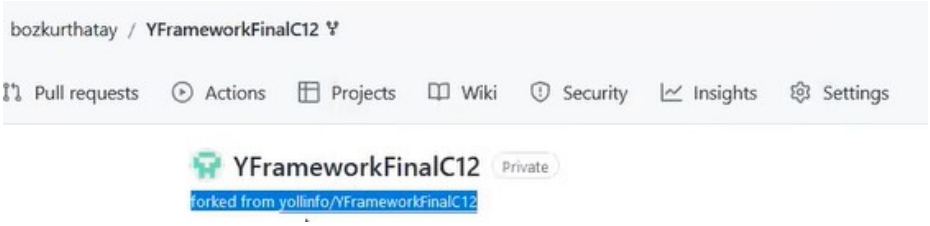
Forking creates a **copy of someone else's repository** under your own GitHub account.

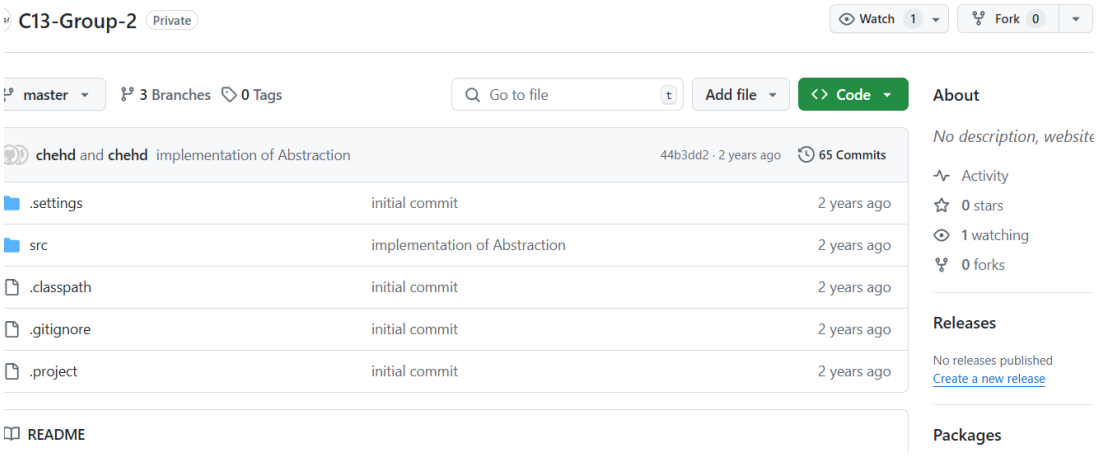
Steps:

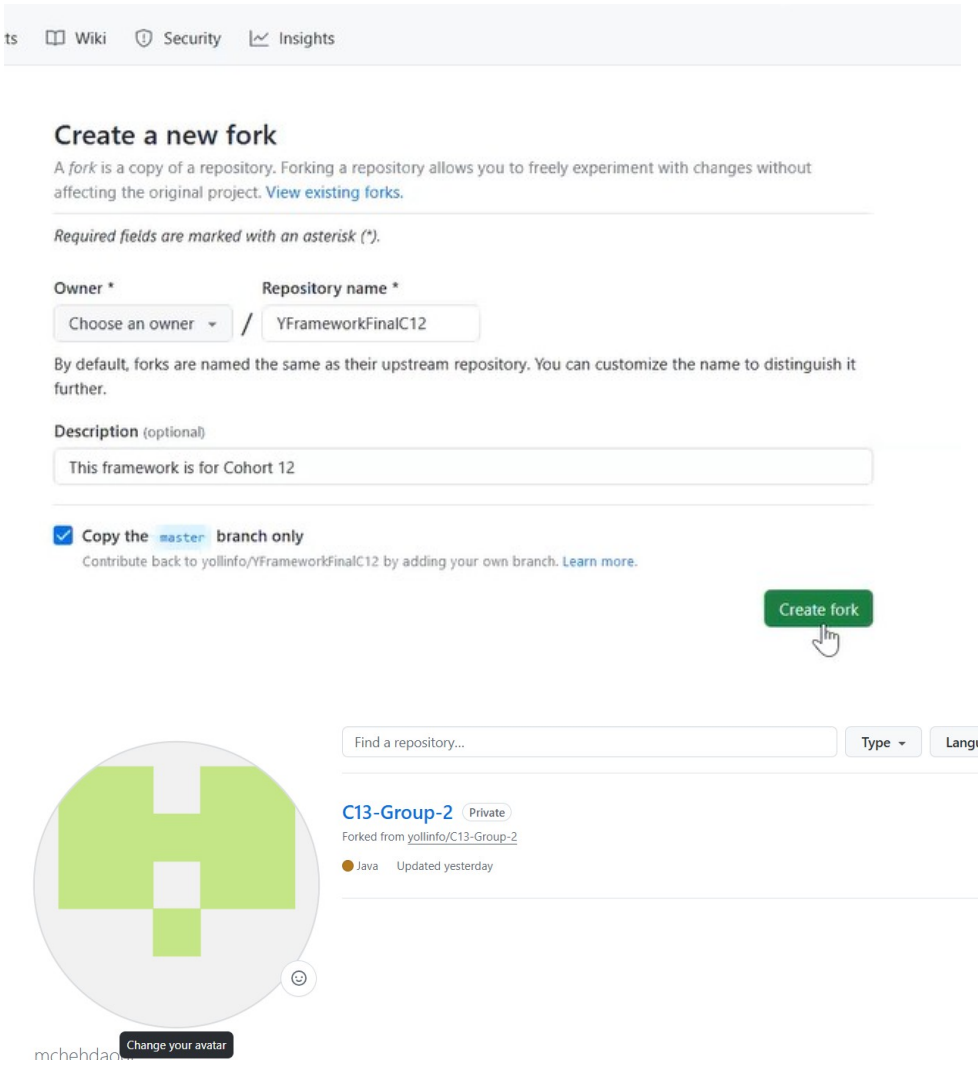
1. **Go to GitHub** and find the repository you want to fork.
   Example: https://github.com/example/repo
2. Click the **"Fork"** button (top-right corner).
3. GitHub will create a **new repository under your account**:

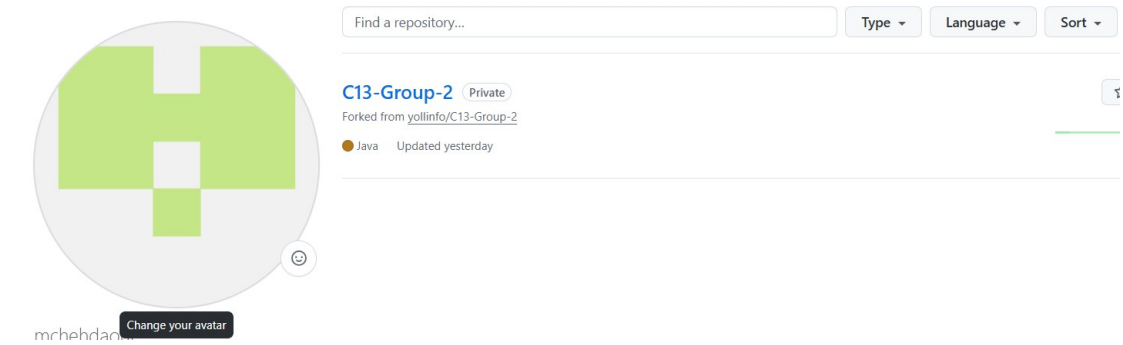
https://github.com/your-username/repo

After Forking, make sure that you see : Forked from ....






ts
Wiki
Security
Insights
Create a new fork
A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. View existing forks.
Required fields are marked with an asterisk (*).
Owner *
Repository name *
Choose an owner
/
YFrameworkFinalC12
By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.
Description (optional)
This framework is for Cohort 12
Copy the master branch only
Contribute back to yollinfo/YFrameworkFinalC12 by adding your own branch. Learn more.
Create fork

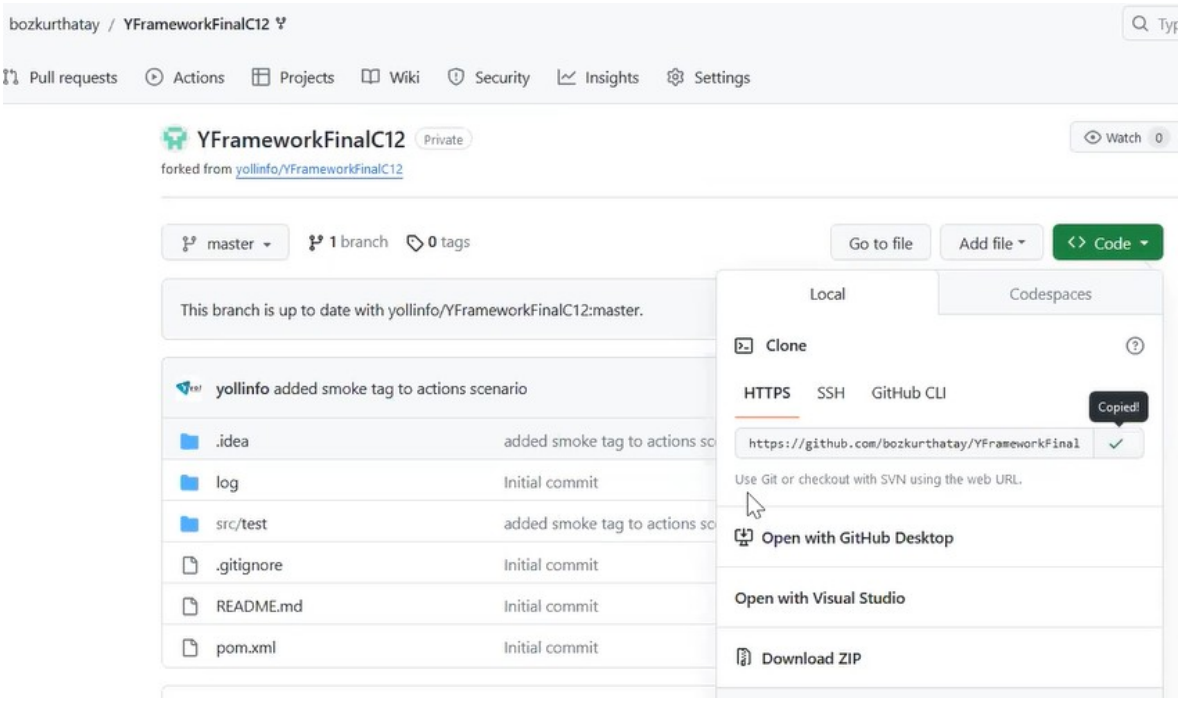


Clone the Forked Repository to Your Local Machine

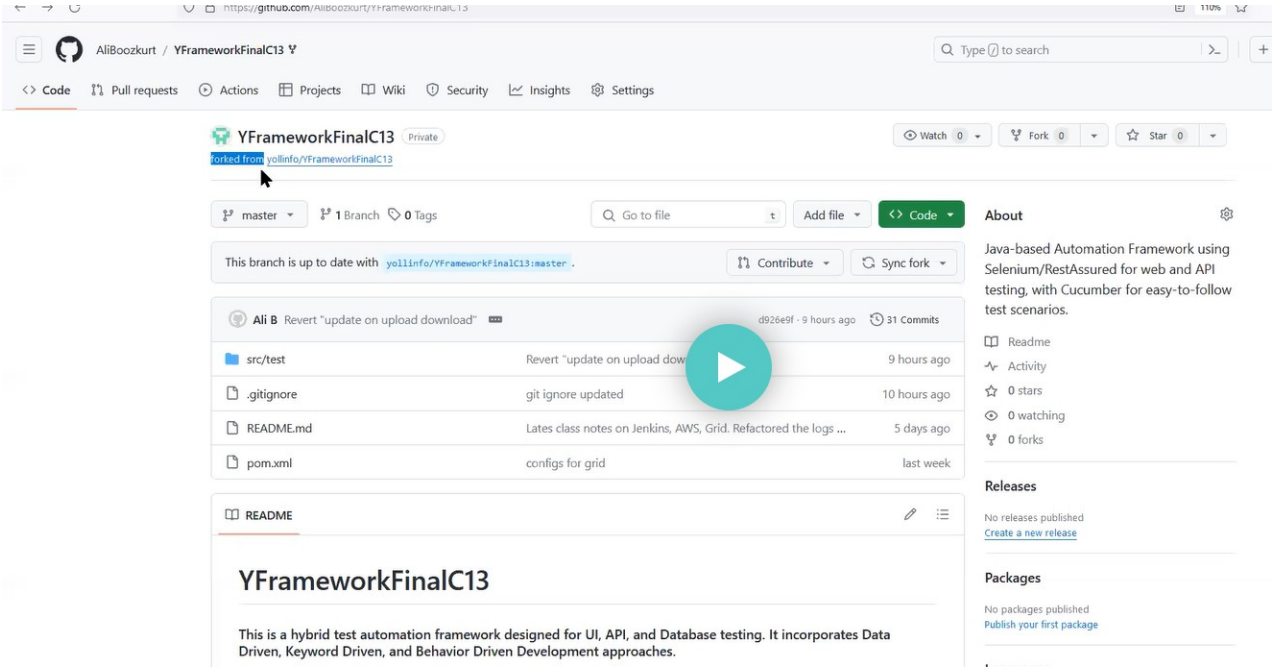
Once you've forked the repo, you need to clone it to your computer.

Steps:

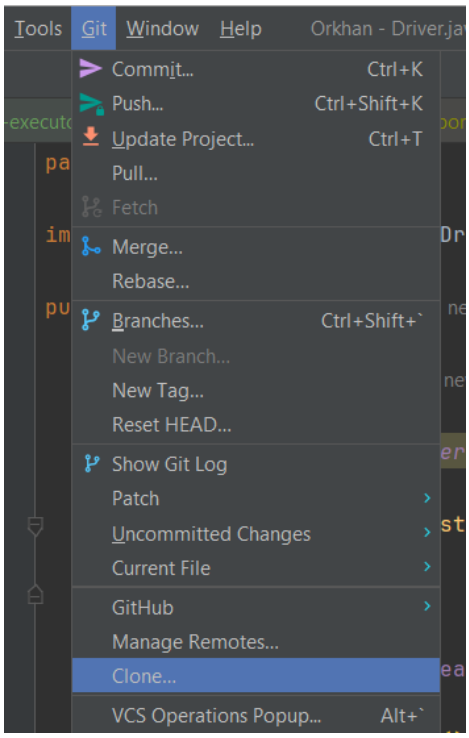
1. **Go to your forked repo** (https://github.com/your-username/repo).
2. Click the "**<>Code**" button (green,) and **copy the URL** (HTTPS or SSH).
3. Open a terminal or command prompt and run



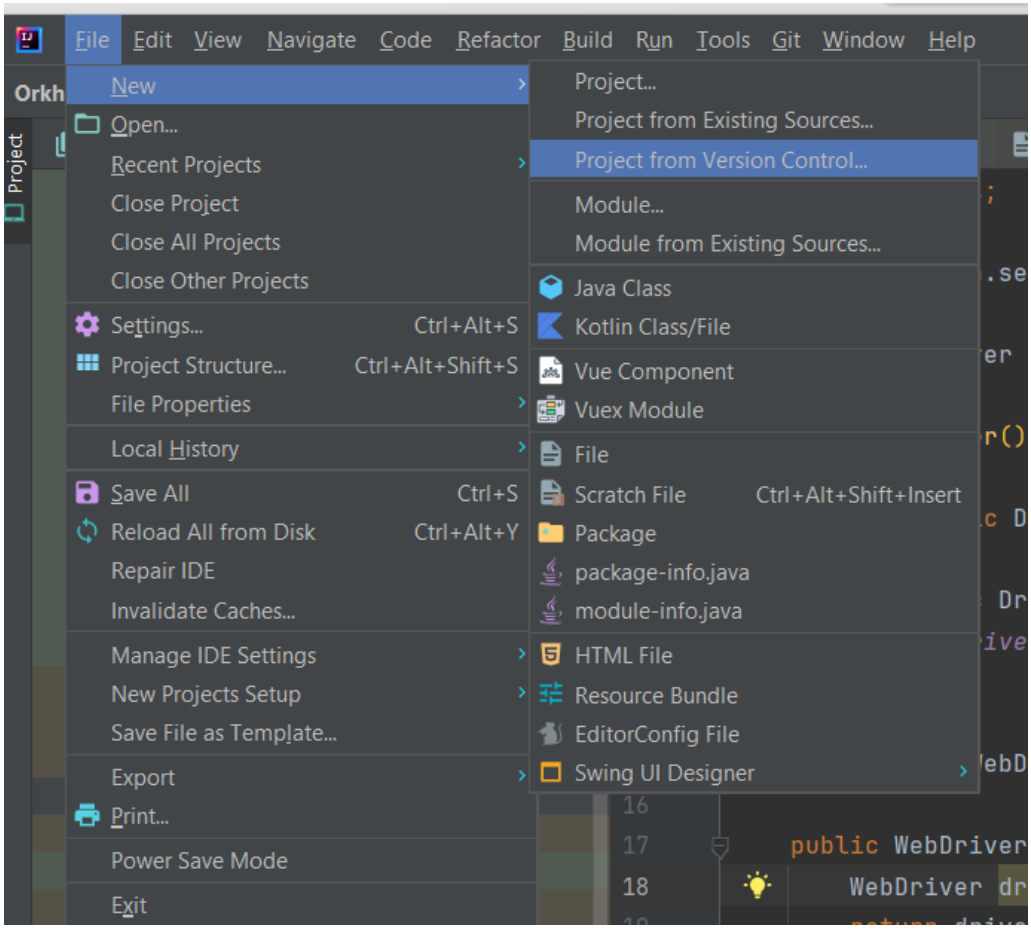
When you are cloning, make sure that you have this line: **Forked from .....**

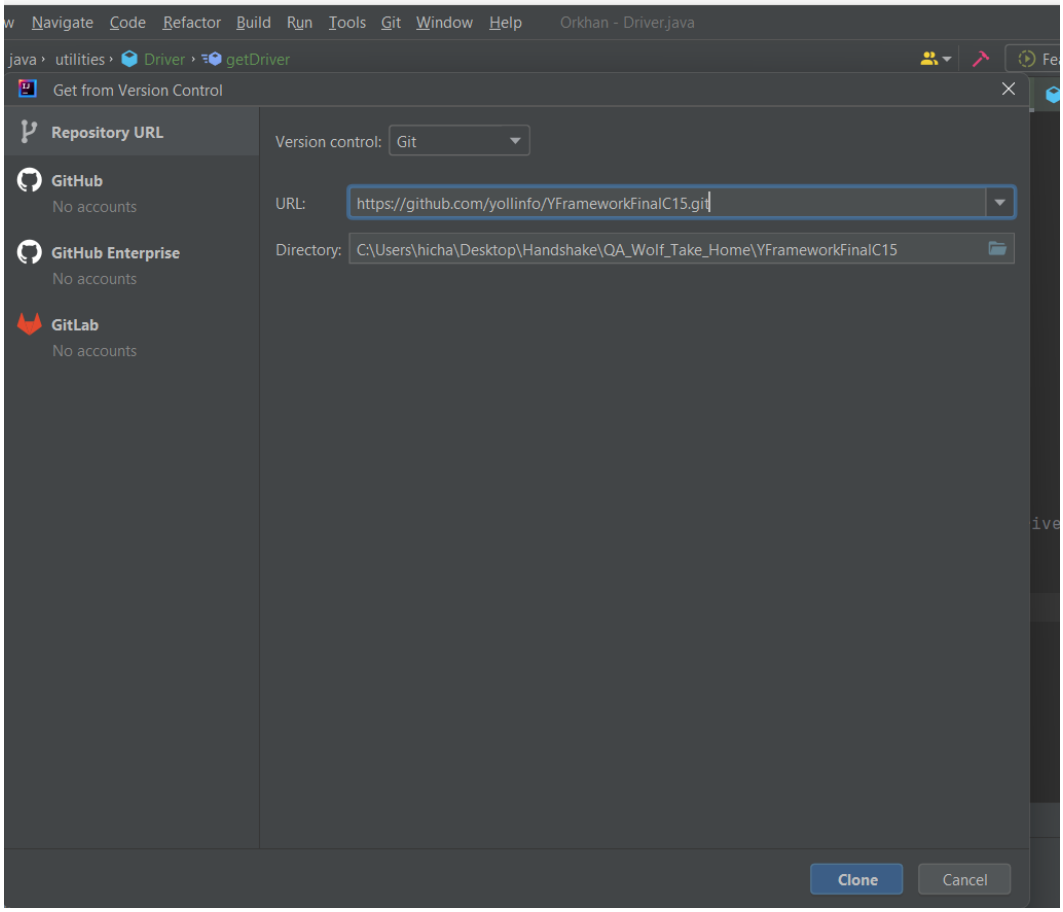


In intellij: Go to Git/ Clone/ Paste the link on URL



Or from File / New/ Project form Version Control





Pull vs Push in Git

`git pull` (Fetch & Merge Updates from Remote to Local)

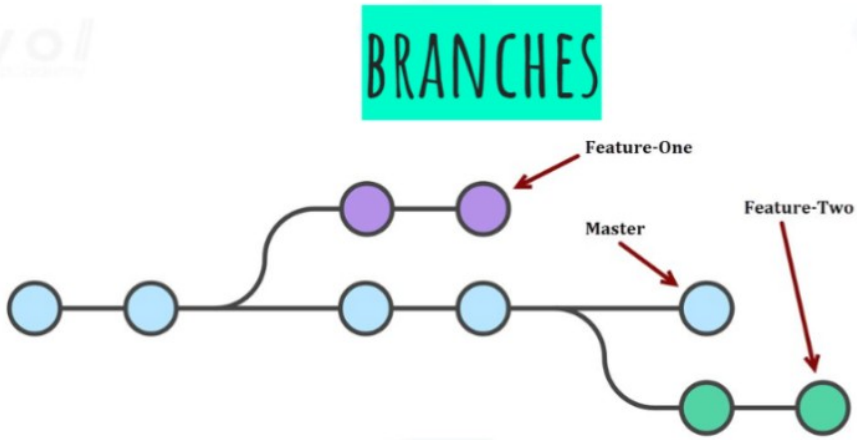
- `git pull` brings changes from a **remote repository** (e.g., GitHub) **into your local branch**.

`git push` (Send Local Changes to Remote Repository)

- git push sends your **local commits** to a **remote repository** (e.g., GitHub).
- Others can now pull your changes into their local repositories.

Key Differences

Feature	git pull	git push
Direction	Remote → Local	Local → Remote
Purpose	Get updates from remote	Upload local changes
Common Use	Before making changes	After committing changes



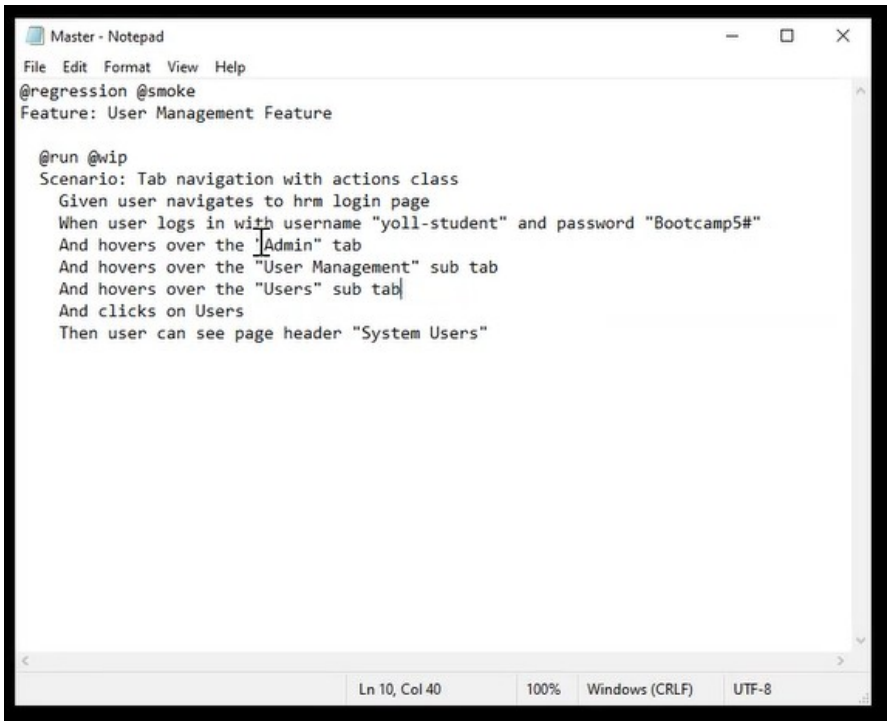
- When you're working on a project, you're going to have a **bunch of different features or ideas** in progress at any given time
- Some of those features are ready to go, and others which are not.
- Branching exists to help you manage this workflow.
- Branches are like **"Save as..."** on a directory.
- Why branch?
  - Consider the utility of "Save as..." for regular files: you **experiment with multiple possibilities while keeping the original safe**.

Branches

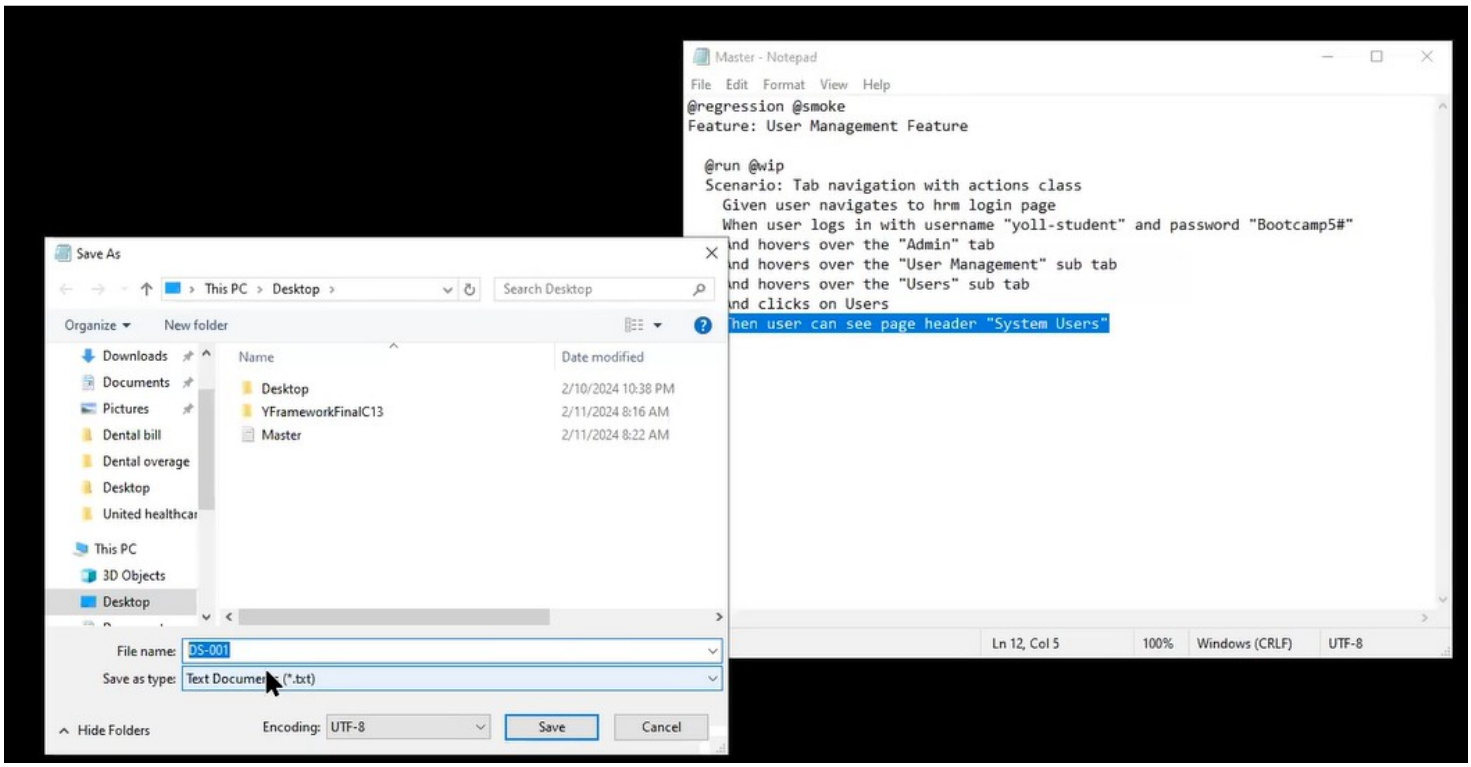
Branches are different files created from ***the master branch*** to keep it seperately without making in changing on it. (there is no pushing om master branch).

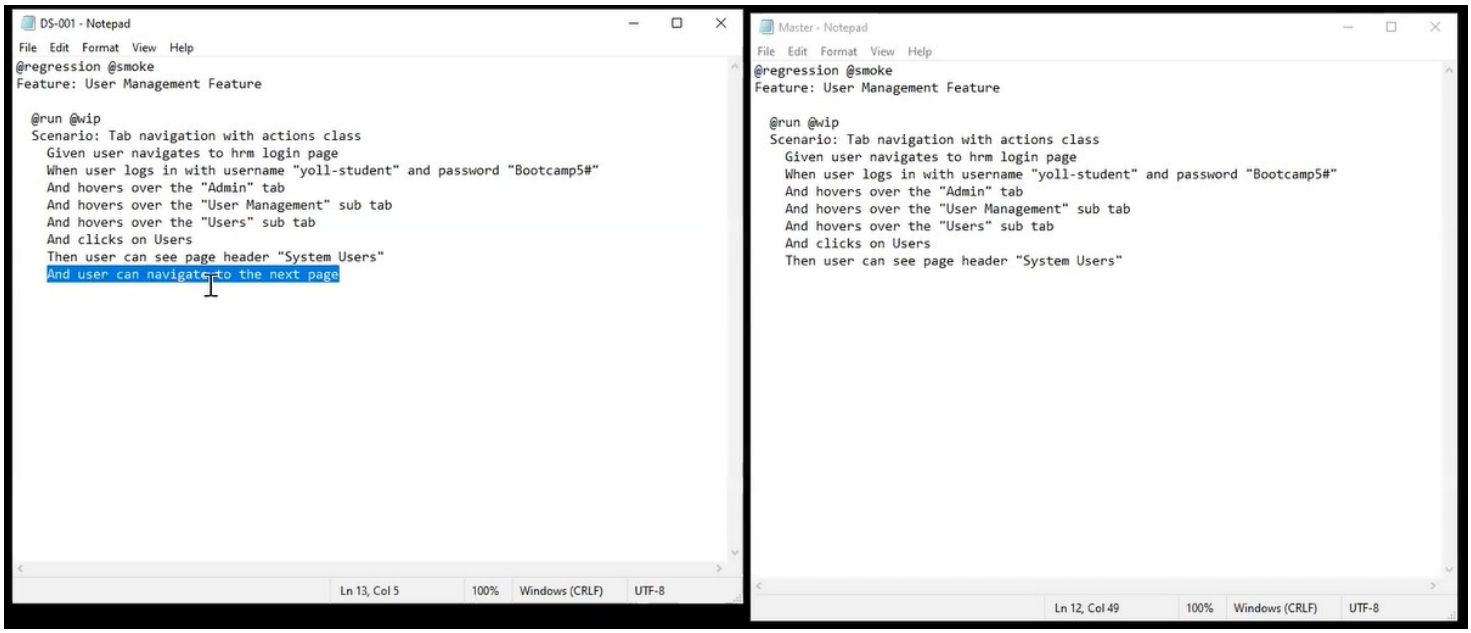
After the ***leader approve*** the changes and update on a file, **then it can be the master branch.**

Example to understand:

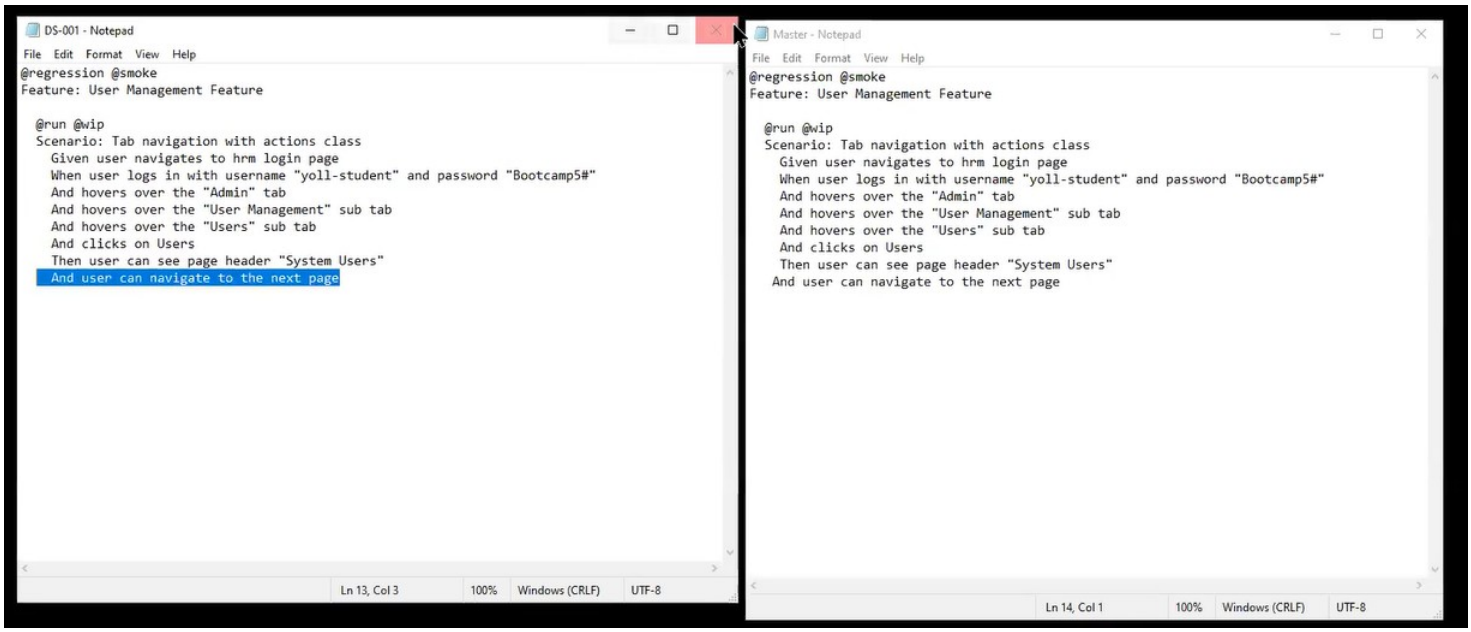

Master - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
Ln 10, Col 40
100%
Windows (CRLF)
UTF-8

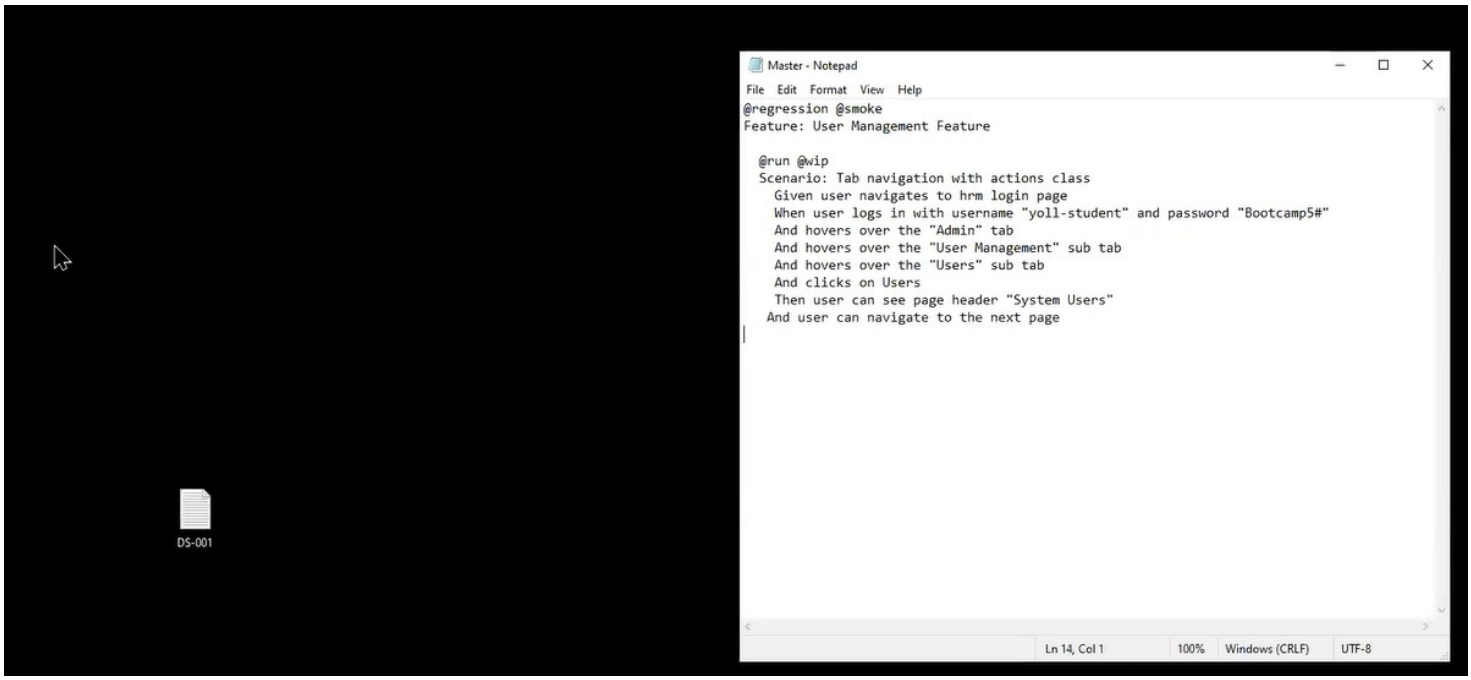
Make a copy: Save AS, and name it. Use the ticket number as Branch name.




DS-001 - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
And user can navigate to the next page
Ln 13, Col 5
100%
Windows (CRLF)
UTF-8
Master - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
Ln 12, Col 49
100%
Windows (CRLF)
UTF-8

After finishing, we merge the files, then we delete the branch


DS-001 - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
And user can navigate to the next page
Ln 13, Col 3
100%
Windows (CRLF)
UTF-8
Master - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
And user can navigate to the next page
Ln 14, Col 1
100%
Windows (CRLF)
UTF-8


Master - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
And user can navigate to the next page
Ln 14, Col 1
100%
Windows (CRLF)
UTF-8
DS-001

What Are Branches in Git?

A **branch** in Git is a separate line of development that allows multiple people to work on different features or fixes **without affecting the main codebase**.

Think of branches as **different workspaces** where you can experiment, make changes, and later merge them into the main code.

🔹 Default Branch (`main` or `master`)

- When you create a new repository, Git automatically creates a default branch (usually named `main` or `master`).
- This branch holds the **stable version** of the project.

🔹 Creating a New Branch

You can create a new branch for a feature or bug fix to keep changes **separate** from the main code.

```
sh
CopyEdit
git branch feature-login
```

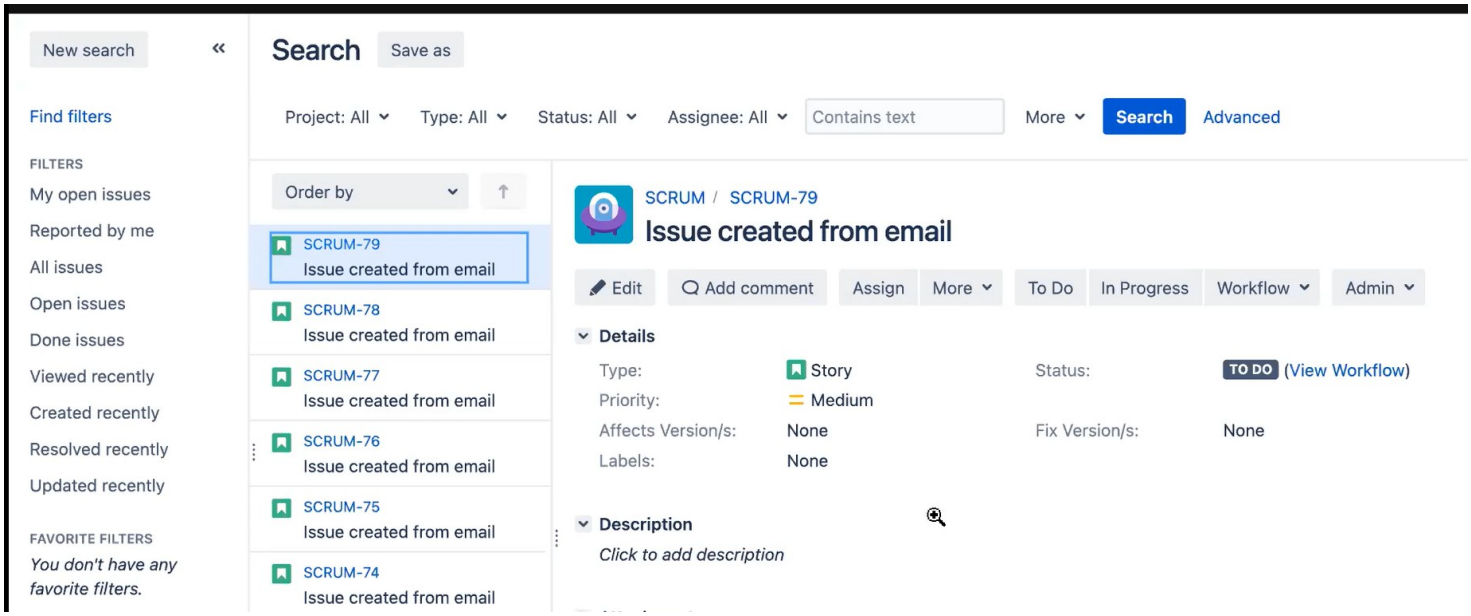
🔹 Creates a new branch named `feature-login`.

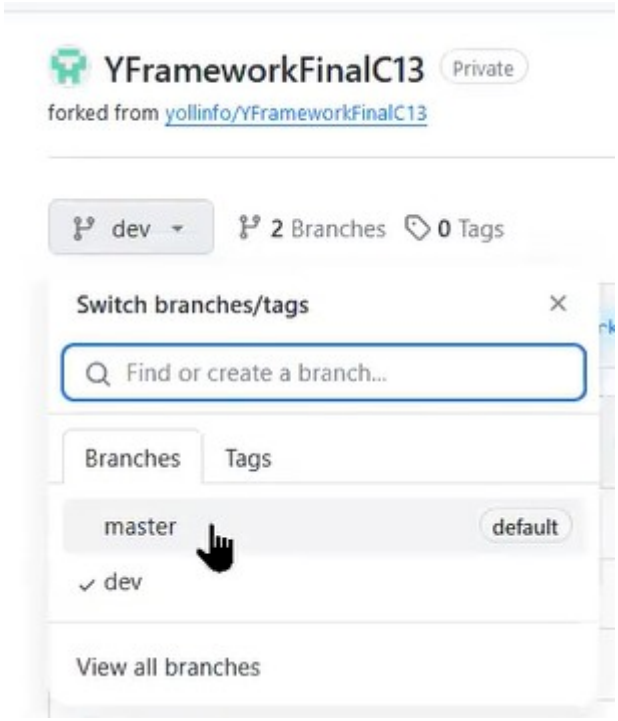
Creating branches from master in GitHub:

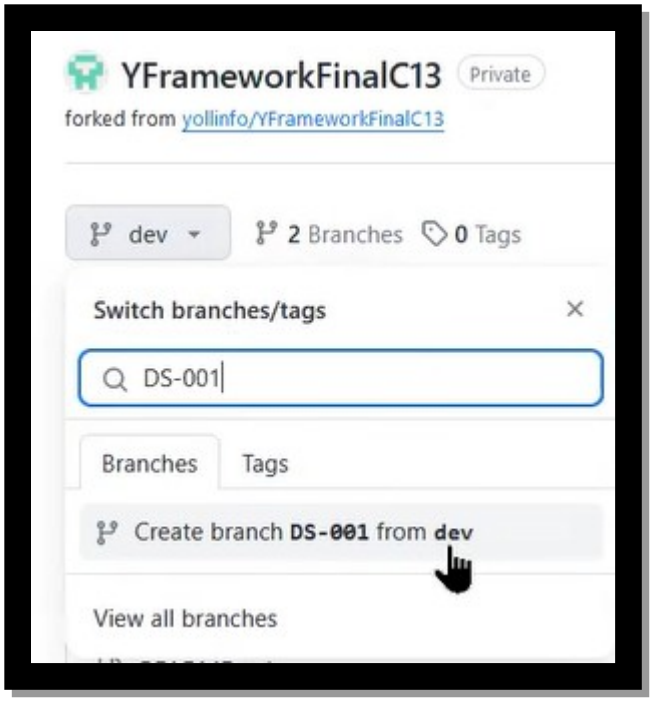
Click on master, write :DS-001. Then click on:

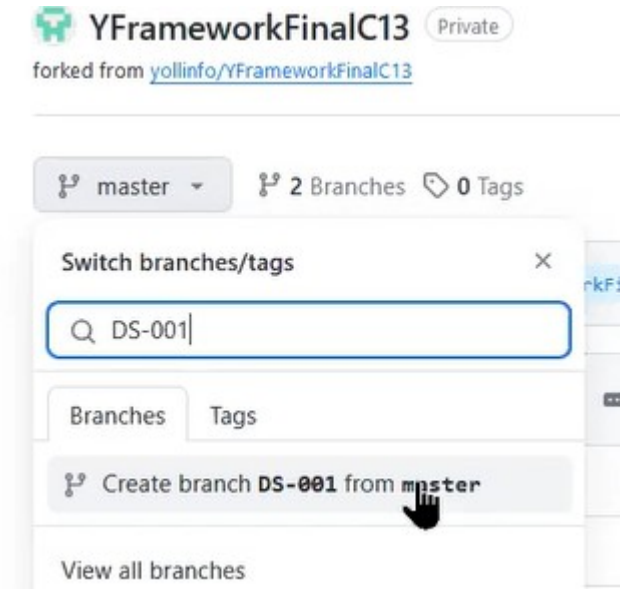
Make sure it is from master not from another branch !

Always use the ticket name like: SCRUM-79

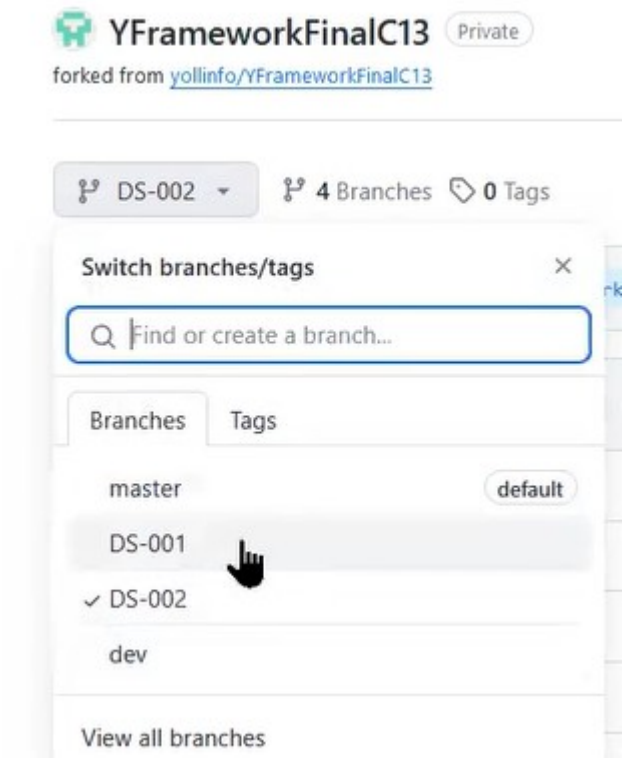






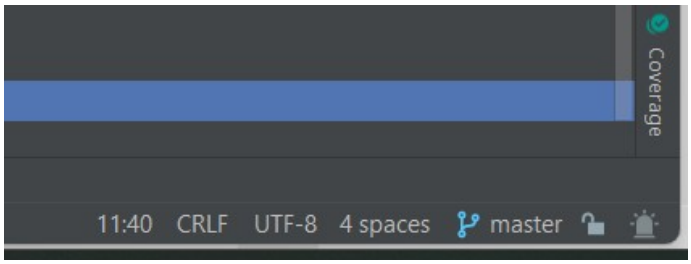


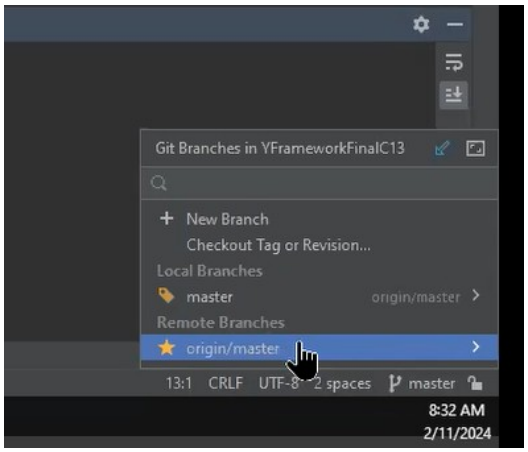
We have FOUR branches:



We need to update intellij because we cannot see the all branches:

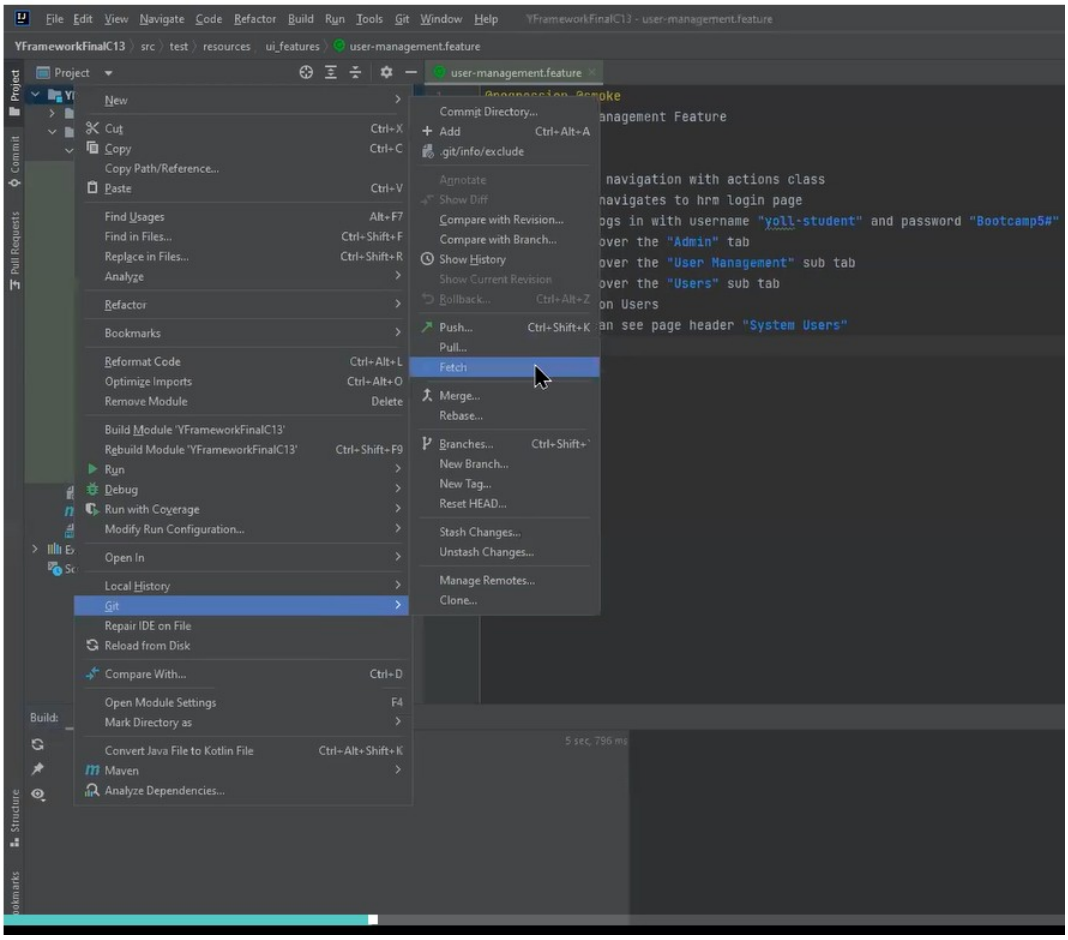
Click on master (Right Bottom)



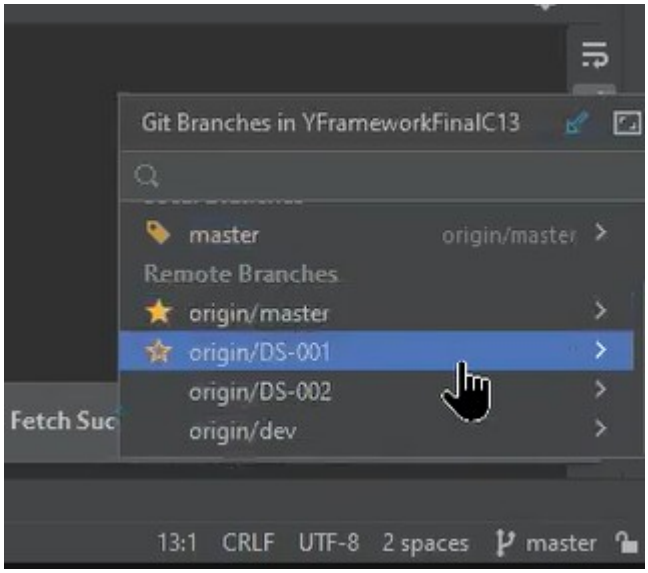


How to update ? Git / Click on Fetch

Fetch is a command used to download changes from a remote repository without automatically merging them into your local branch.

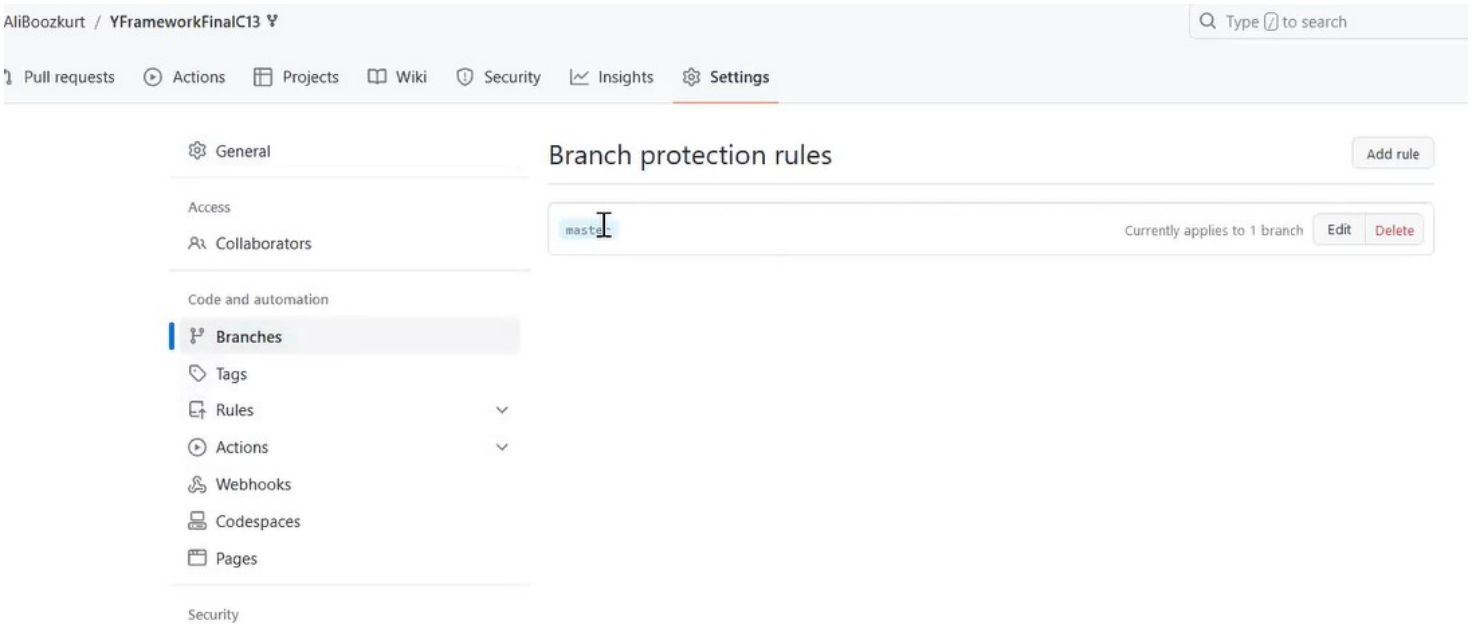


Scroll Down to see the update:

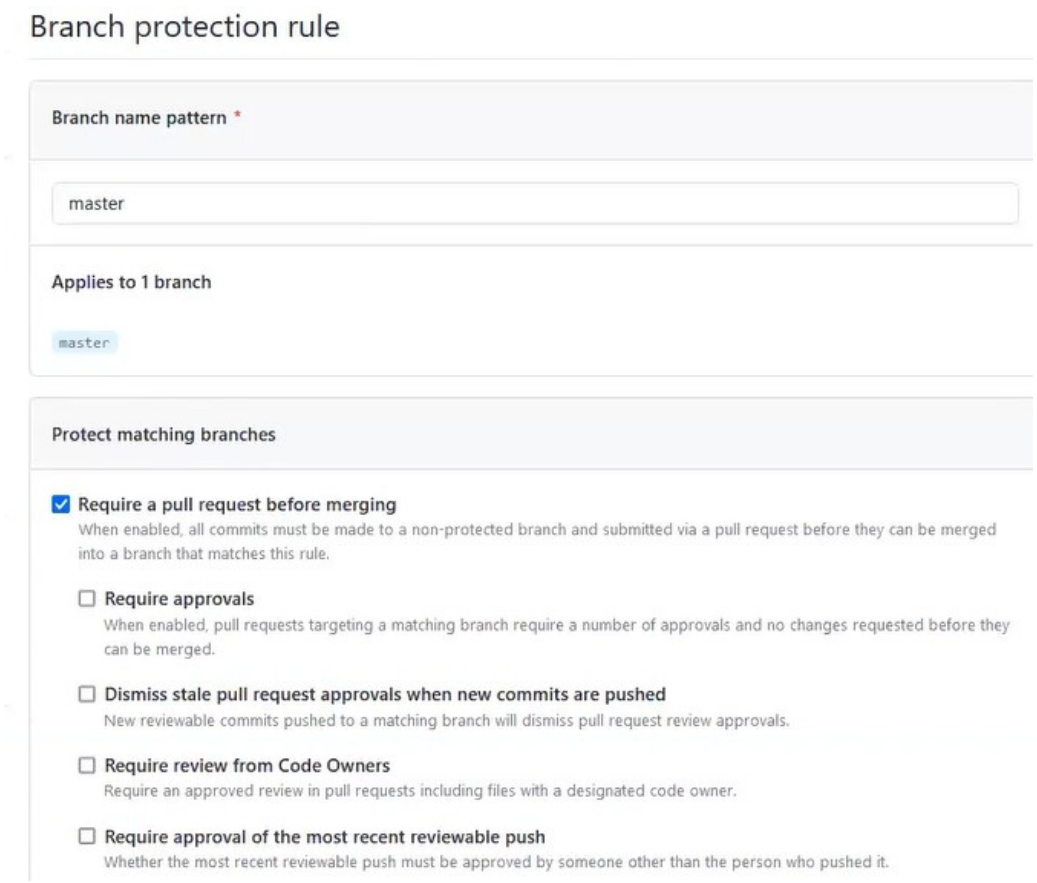


To protect the master branch and the other branches:

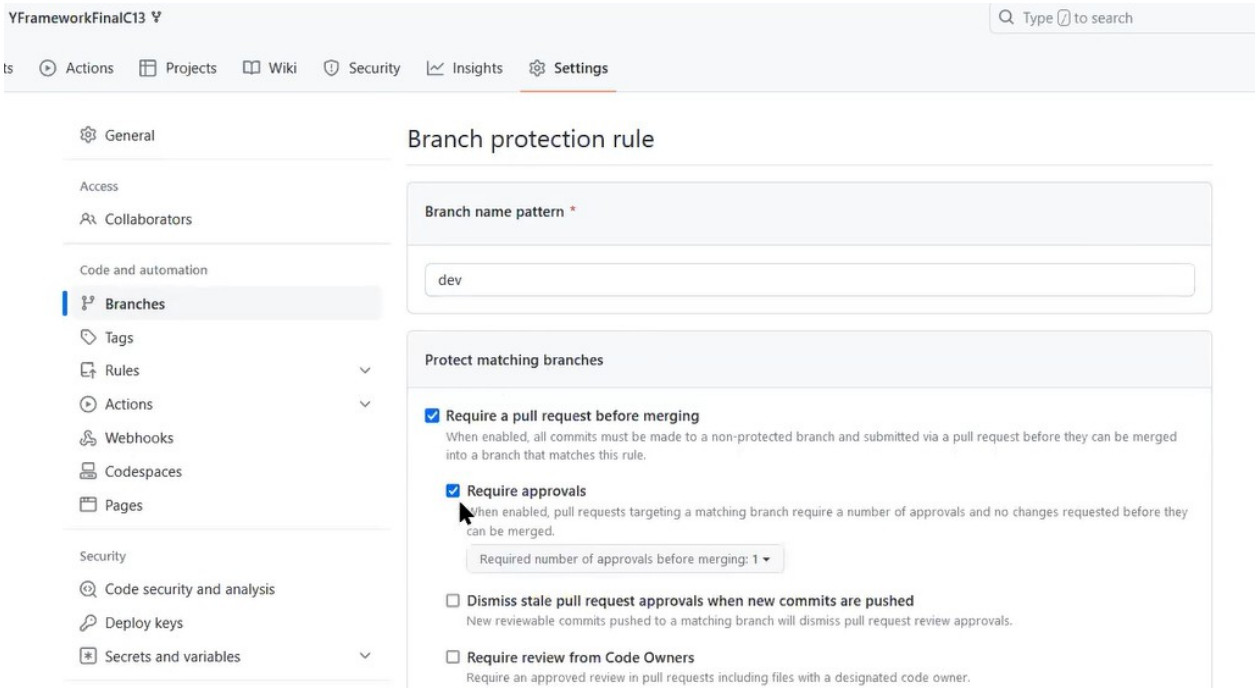
Go to Settings/ branch/ Edit



For master branch


Branch protection rule
Branch name pattern *
master
Applies to 1 branch
master
Protect matching branches
Require a pull request before merging
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.
Require approvals
When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.
Dismiss stale pull request approvals when new commits are pushed
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.
Require review from Code Owners
Require an approved review in pull requests including files with a designated code owner.
Require approval of the most recent reviewable push
Whether the most recent reviewable push must be approved by someone other than the person who pushed it.

For Dev branch:


YFrameworkFinalC13
Type / to search
Actions
Projects
Wiki
Security
Insights
Settings
General
Access
Collaborators
Code and automation
Branches
Tags
Rules
Actions
Webhooks
Codespaces
Pages
Security
Code security and analysis
Deploy keys
Secrets and variables
Branch protection rule
Branch name pattern *
dev
Protect matching branches
Require a pull request before merging
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.
Require approvals
When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.
Required number of approvals before merging: 1
Dismiss stale pull request approvals when new commits are pushed
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.
Require review from Code Owners
Require an approved review in pull requests including files with a designated code owner.

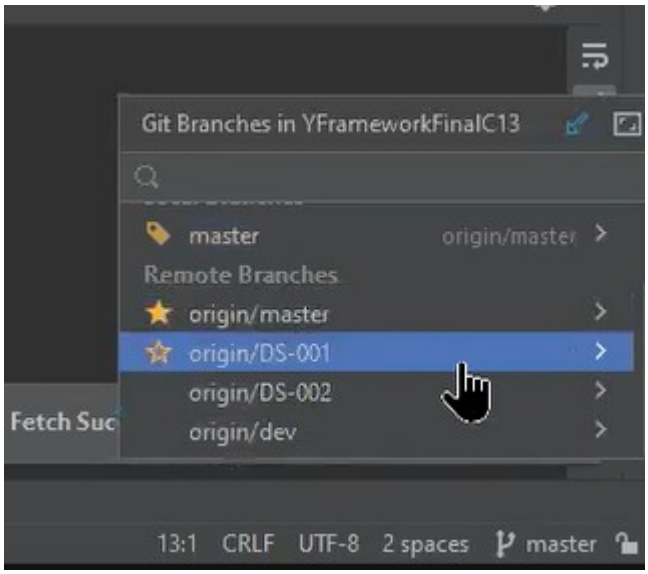
For any update : Git / Click on Fetch

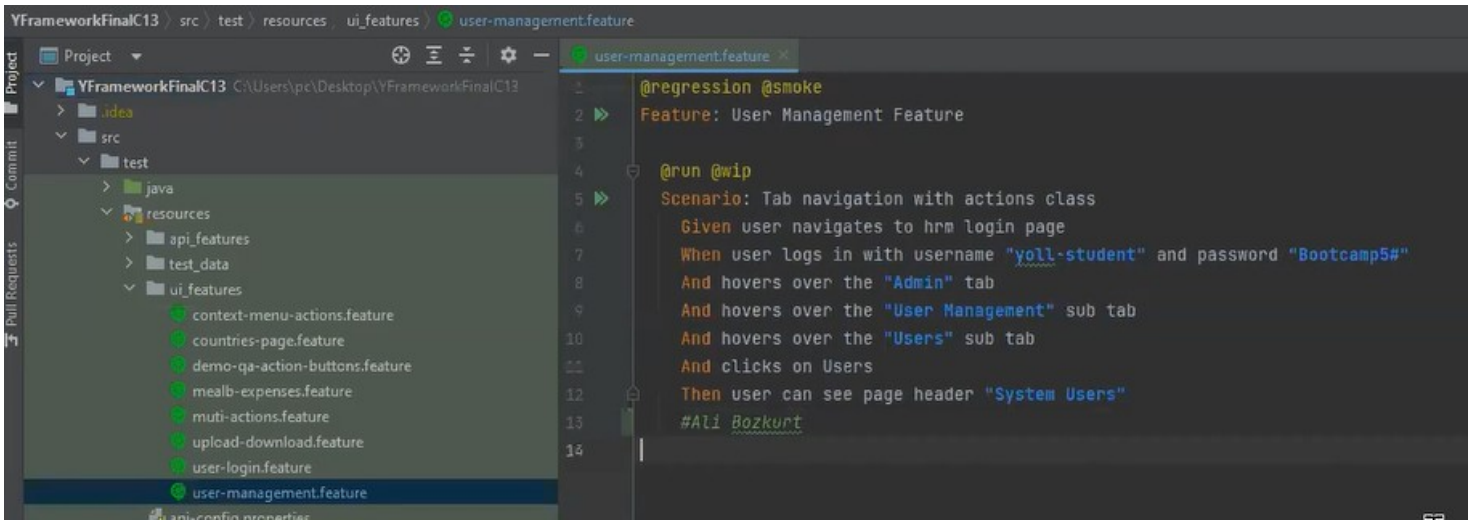
Task:

Choose the master branch by clicking on right bottom, and add a line on user-management.feature:

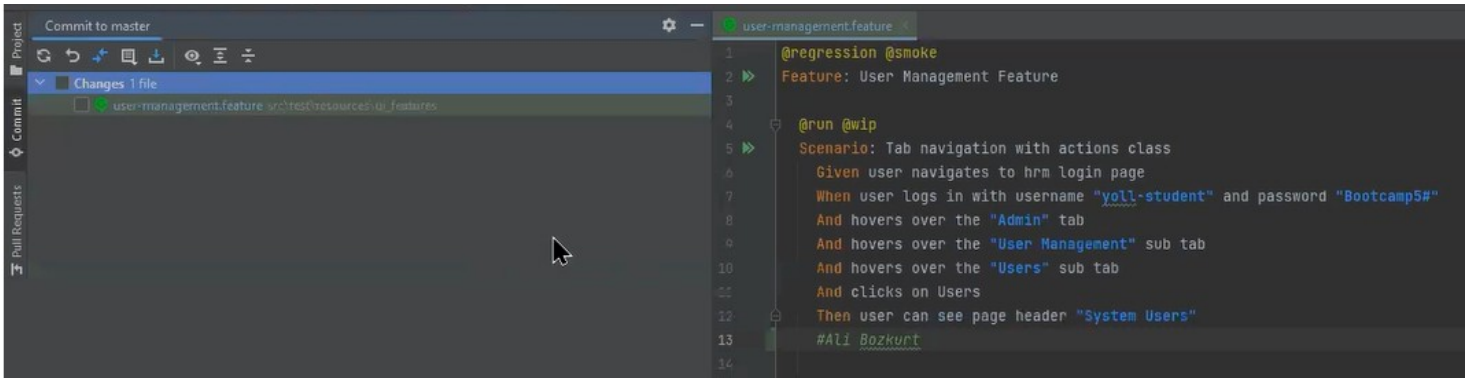
#Ali Buskort

Then save it

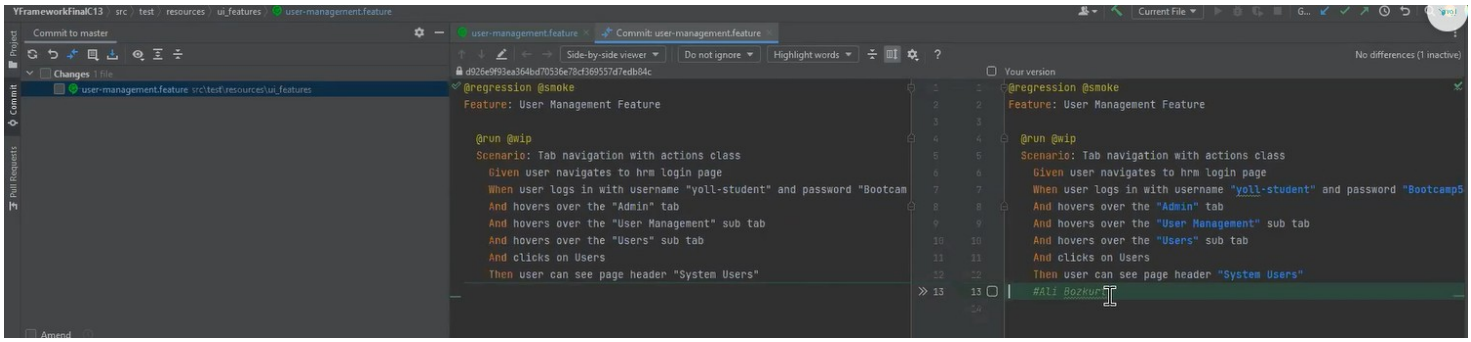




To see the changes, click on commit


Commit to master
Changes 1 file
user-management.feature src\test\resources\ui_features
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Ali Bozkurt

Double click on user-management.feature to see the changes before and after:

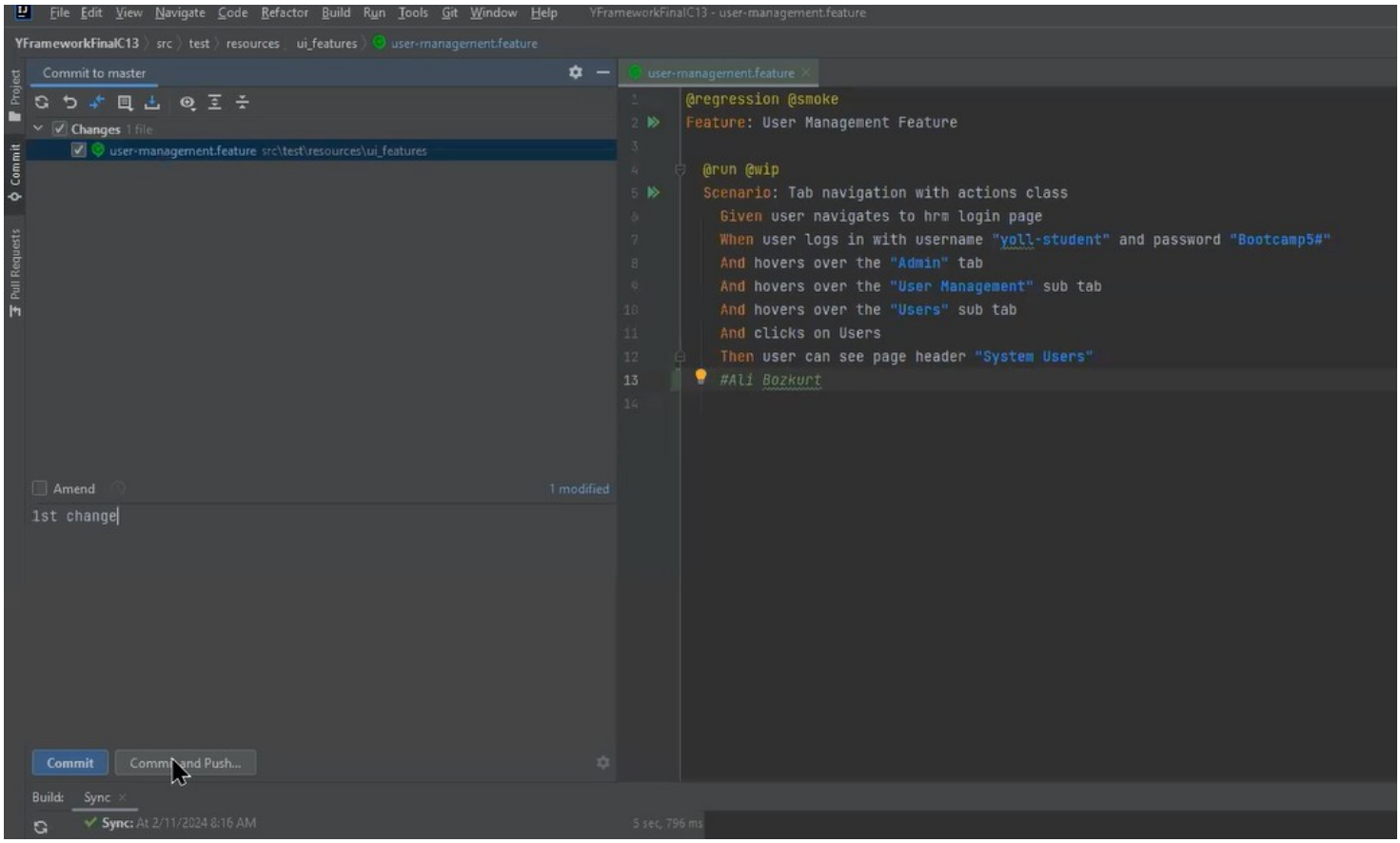

Commit to master
Changes 1 file
user-management.feature src\test\resources\ui_features
Commit: user-management.feature
Side-by-side viewer
Do not ignore
Highlight words
No differences (1 inactive)
Your version
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Ali Bozkurt

How to push ?

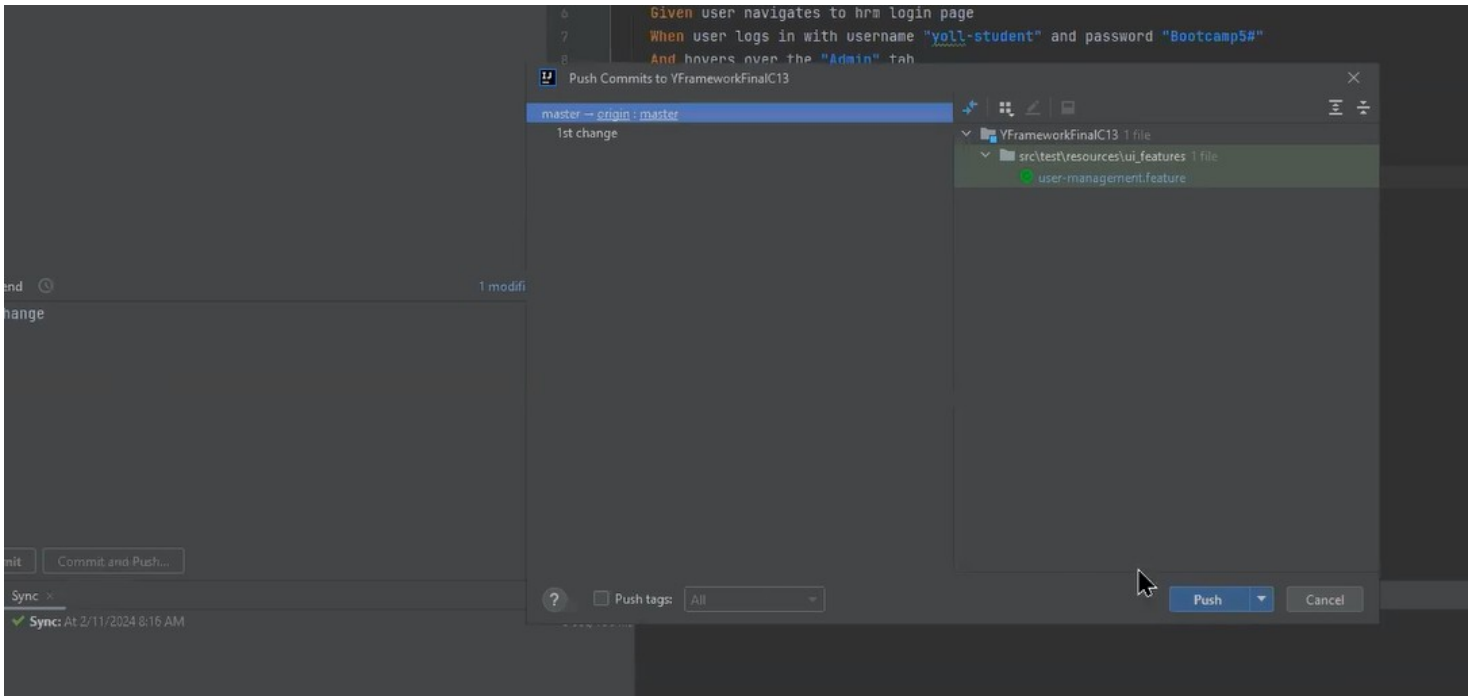
Select user-management.feature

Add comment: 1st change.

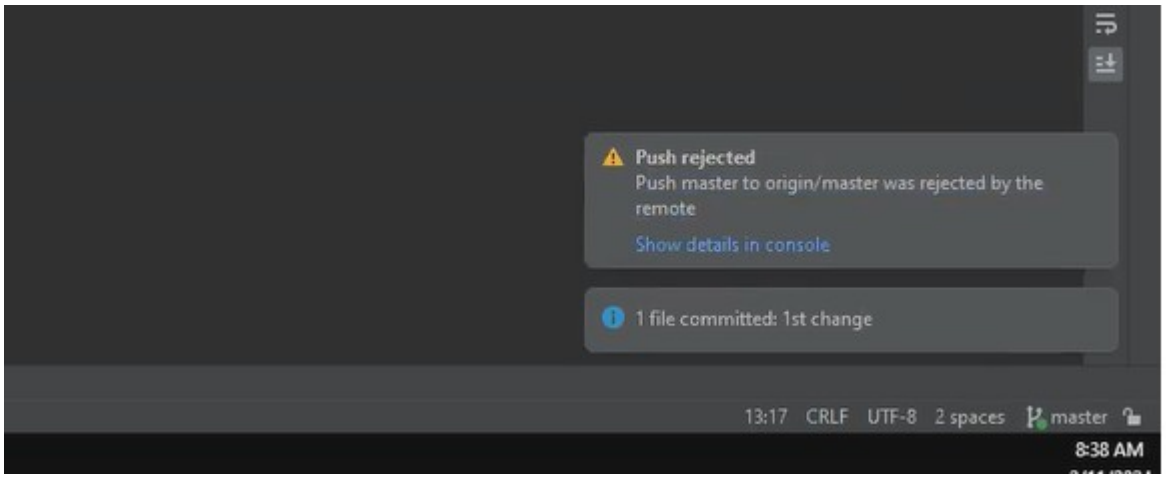
Click on Commit and Push


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
YFrameworkFinalC13 - user-management.feature
YFrameworkFinalC13 src test resources ui_features user-management.feature
Commit to master
Changes 1 file
user-management.feature src\test\resources\ui_features
Amend
1 modified
1st change
Commit
Commit and Push...
Build: Sync
Sync: At 2/11/2024 8:16 AM
5 sec, 796 ms
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Ali Bozkurt

Click on Push



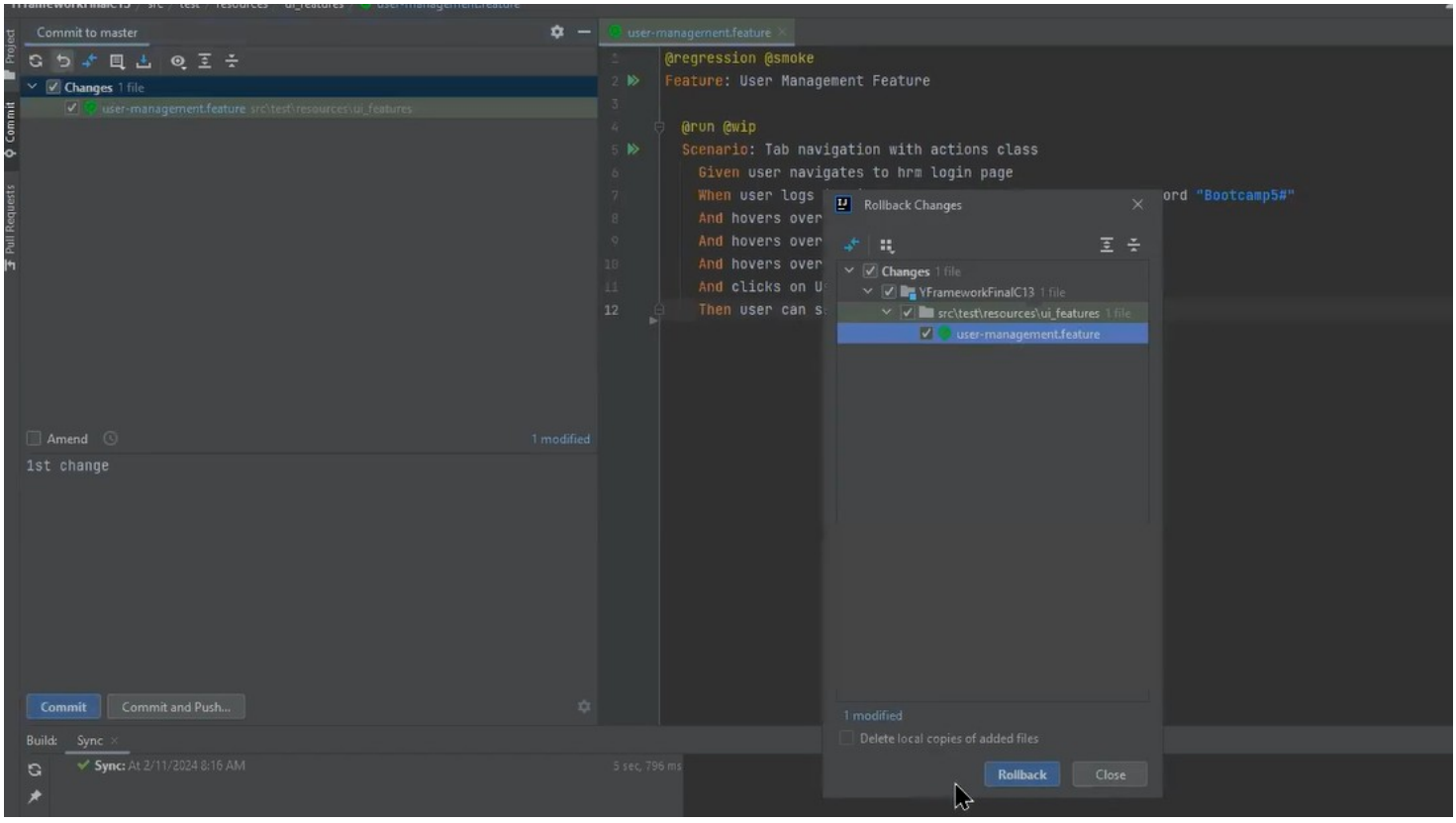
If the branch is protected, it will be rejected to be pushed. It is supposed to be controled and reviewed.


Push rejected
Push master to origin/master was rejected by the remote
Show details in console
1 file committed: 1st change
13:17 CRLF UTF-8 2 spaces master
8:38 AM

How to roll back (Cancel the change without doing any push)

Select user-management.feature

Click on Rollback.

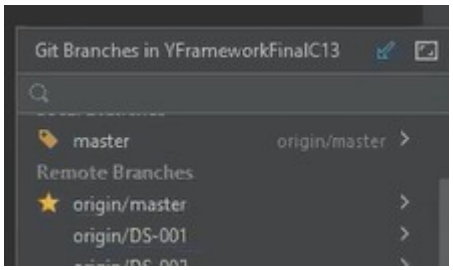


Reset it:

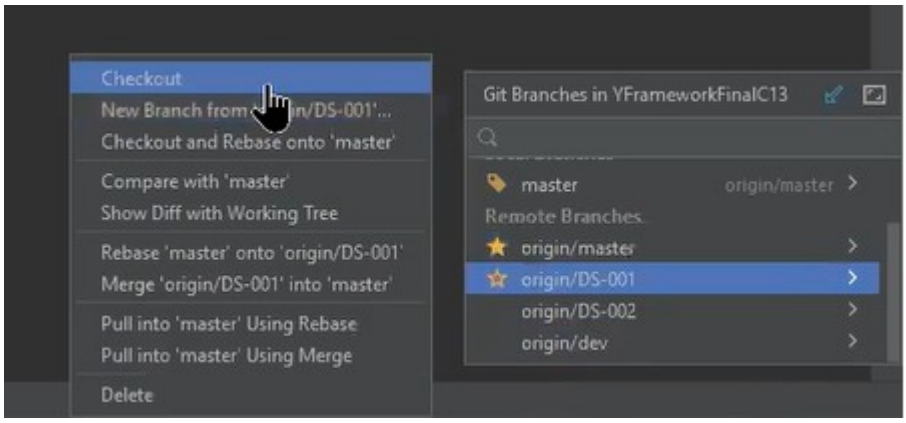
```
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git reset --hard origin/master
HEAD is now at d926e9f Revert "update on upload download"
PS C:\Users\pc\Desktop\YFrameworkFinalC13>
```

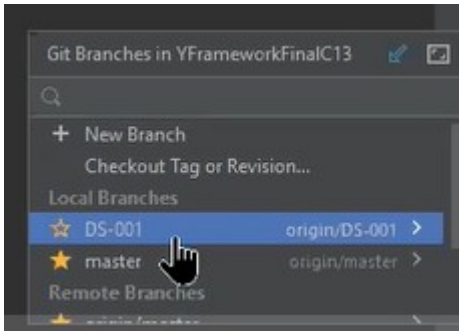
How to switch from branch to branch ? Checkout

Now, we are on master branch, it has a star before the name.

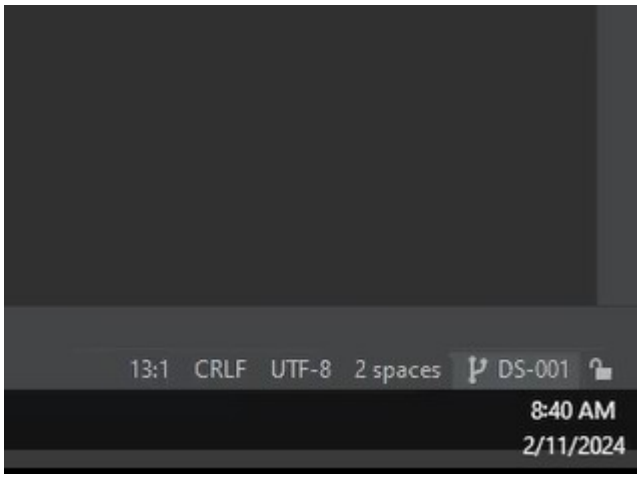


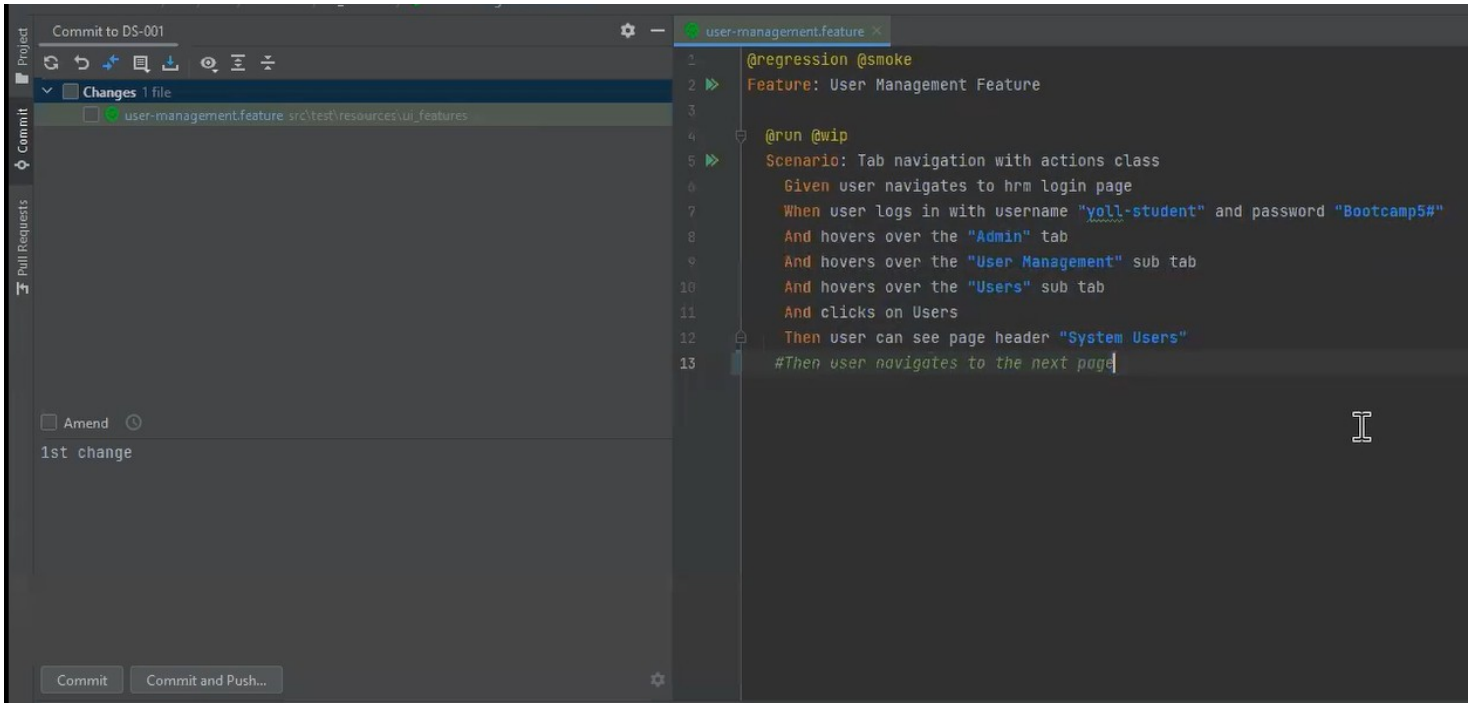
In order to switch to DS-001, click on it, then Checkout





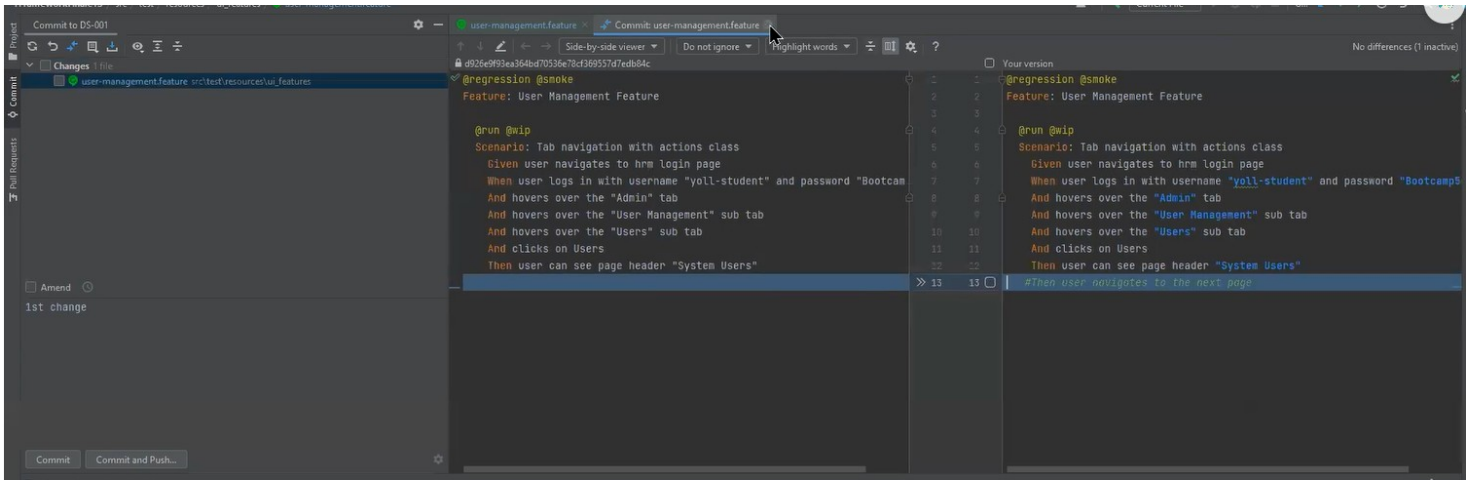
We can see that we are at the right branch:



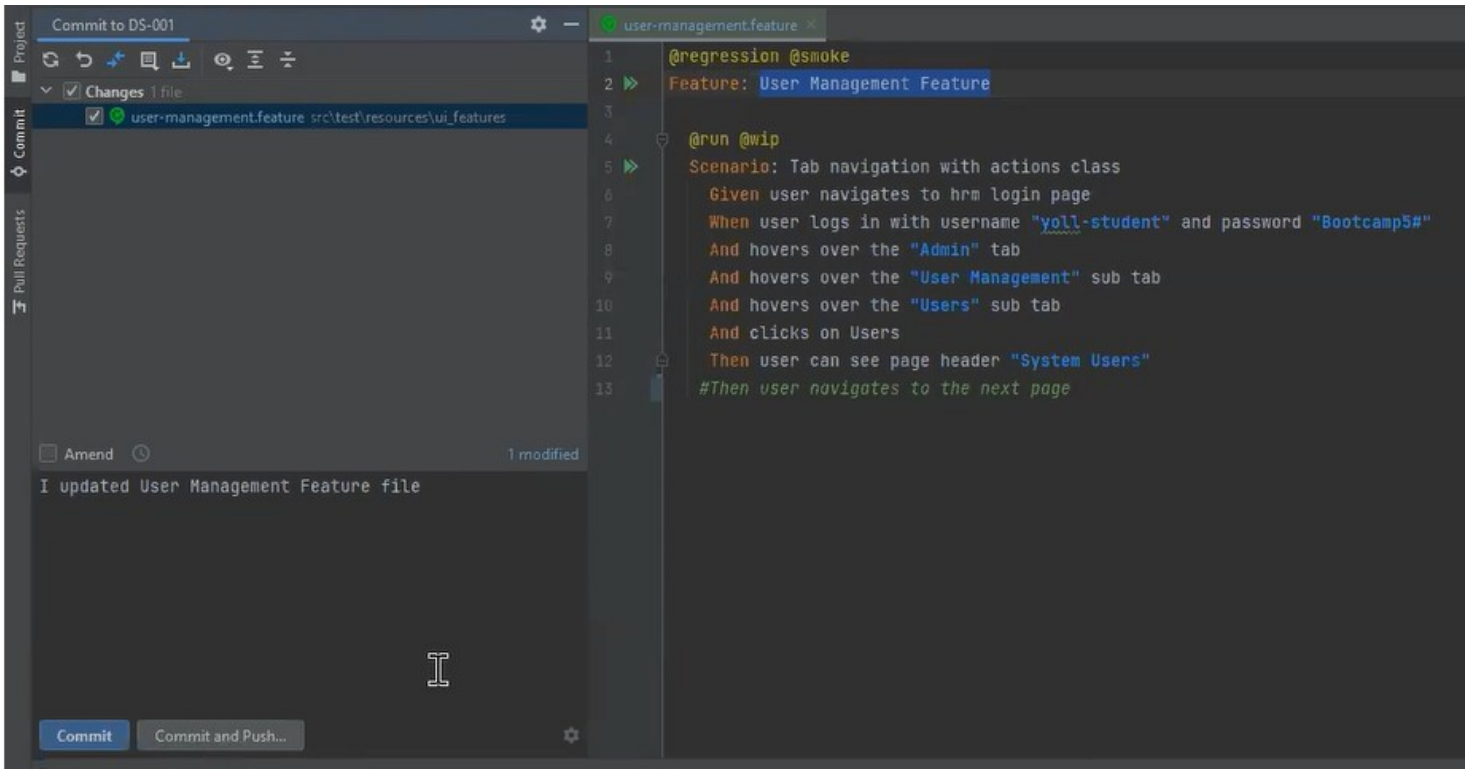

Commit to DS-001
Changes 1 file
user-management.feature src\test\resources\ui_features
Amend
1st change
Commit
Commit and Push...
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page

Before and after in the branch DS-001

ALWAYS DOUBLE CHECK

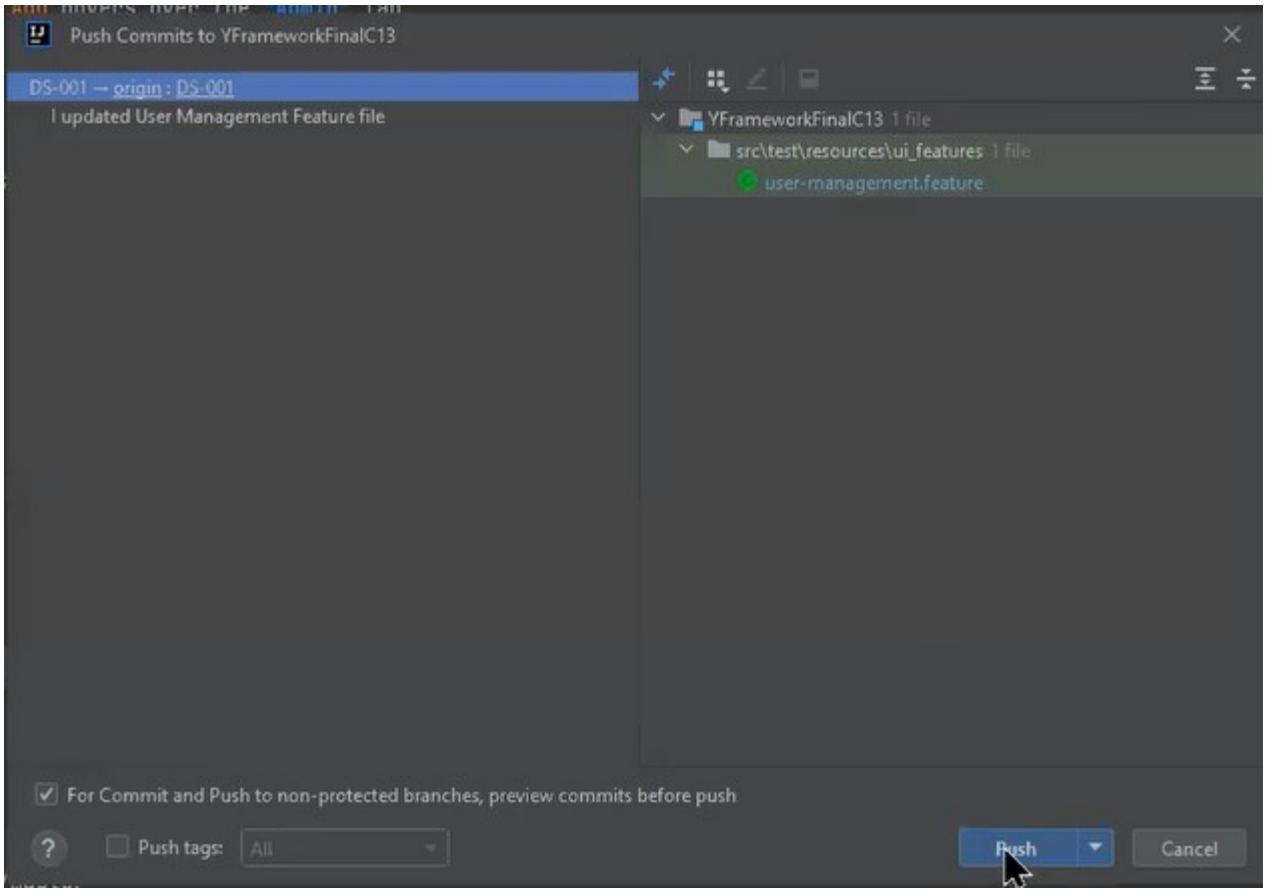


The comment must be meaningful: I updated User Management Feature File

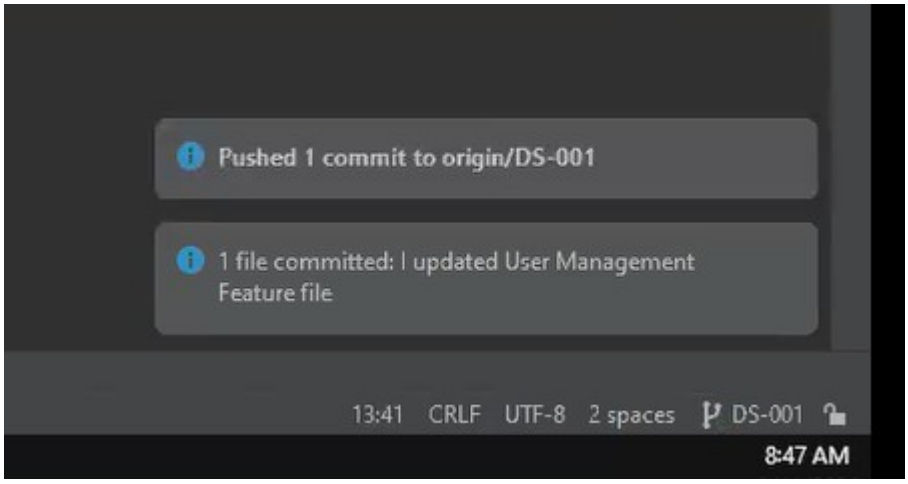

Commit to DS-001
Changes 1 file
user-management.feature src\test\resources\ui_features
Amend
1 modified
I updated User Management Feature file
Commit
Commit and Push...
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page

Then Click on commit and Push

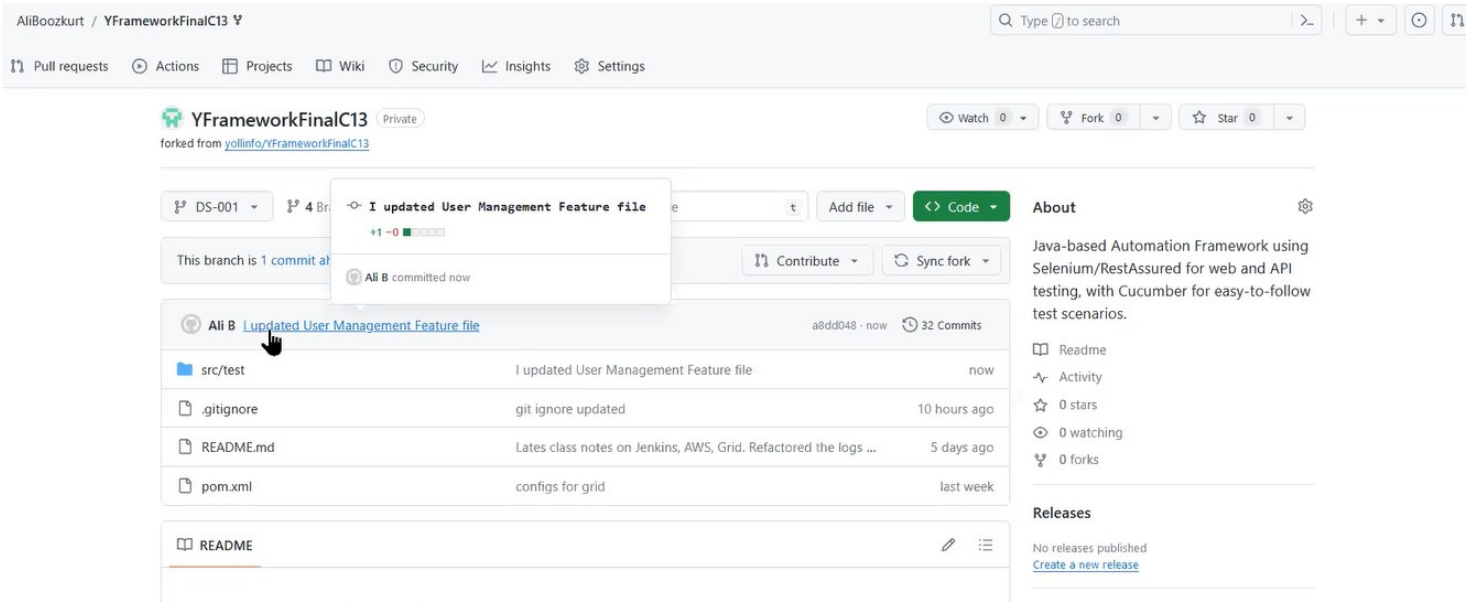
Push

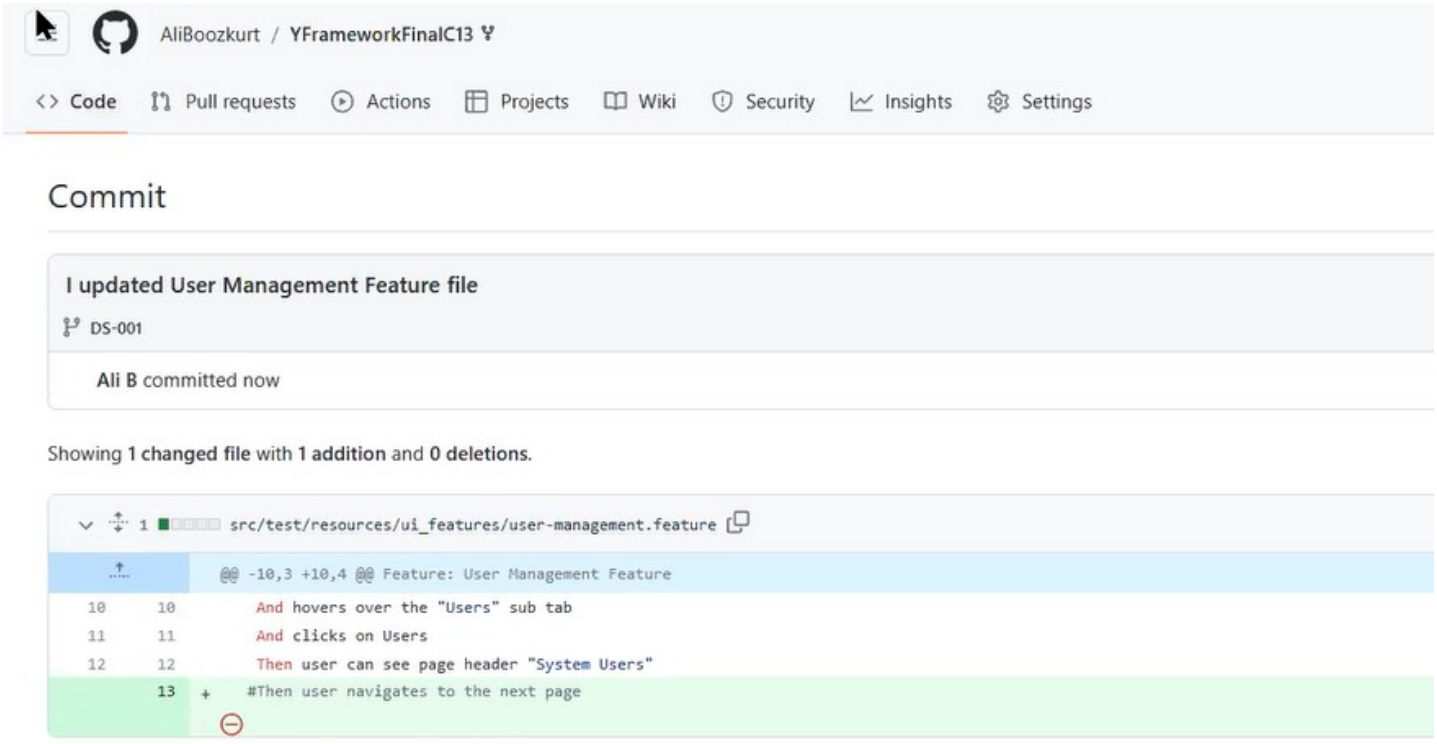


We can see the messages: at this time, the branch DS-001 is not protected as the master branch.


Pushed 1 commit to origin/DS-001
1 file committed: I updated User Management Feature file
13:41 CRLF UTF-8 2 spaces DS-001
8:47 AM

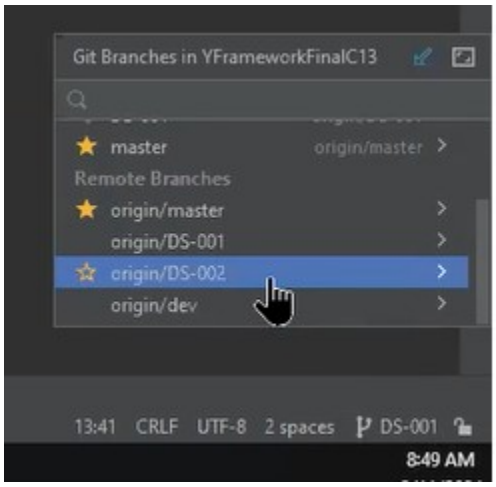
Result on GitHub:



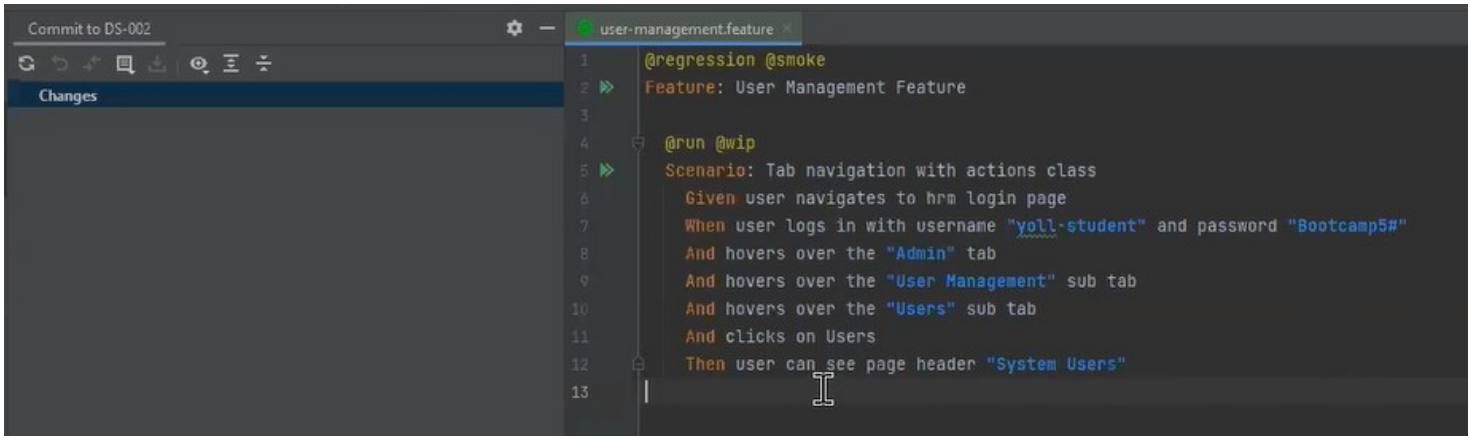


The change can be only seen on DS-001 not on the other branches.

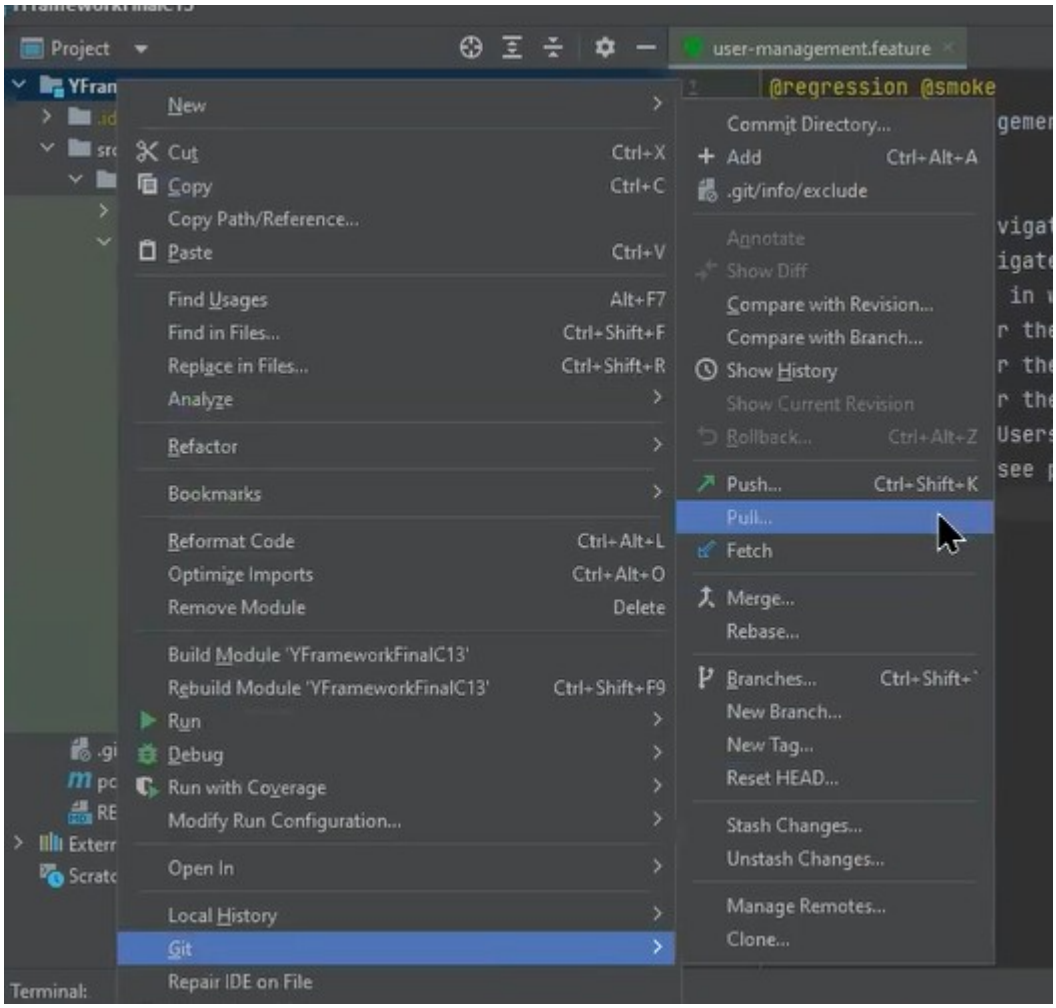
Go to DS-002

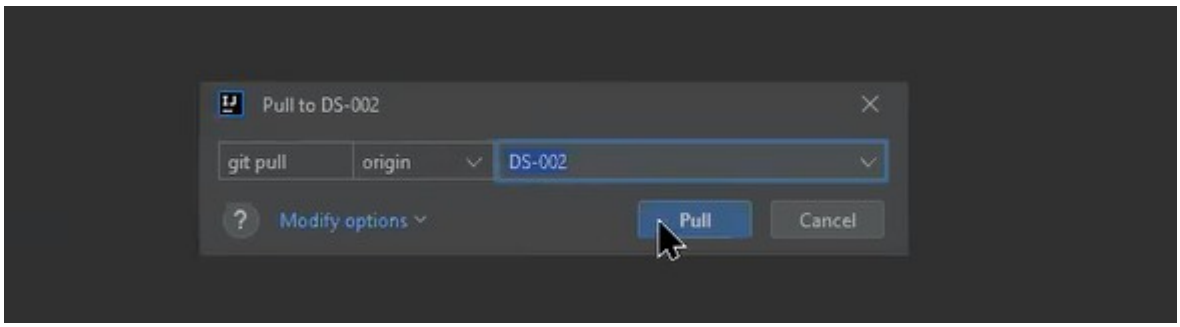


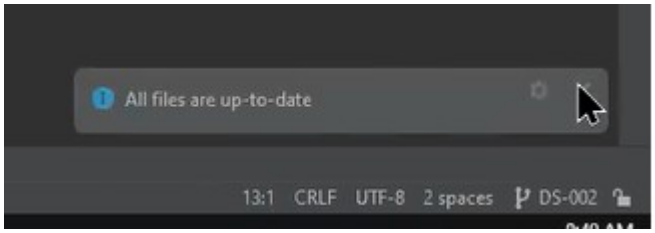
There is no change on DS-002 !


Commit to DS-002
Changes
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"

How to pull ?







🔄 Switching to a Branch

```
sh
CopyEdit
git checkout feature-login
```

or (in newer Git versions)

```
sh
CopyEdit
git switch feature-login
```

 Moves you to the `feature-login` branch, allowing you to work on it.

Creating & Switching in One Command

```
sh
CopyEdit
git checkout -b feature-login
```

or

```
sh
CopyEdit
git switch -c feature-login
```

 Creates and switches to `feature-login`.

Merging a Branch into `main`

After finishing your work, you merge it back into the main branch:

```
sh
CopyEdit
git checkout main
git merge feature-login
```

 Merges `feature-login` into `main`.

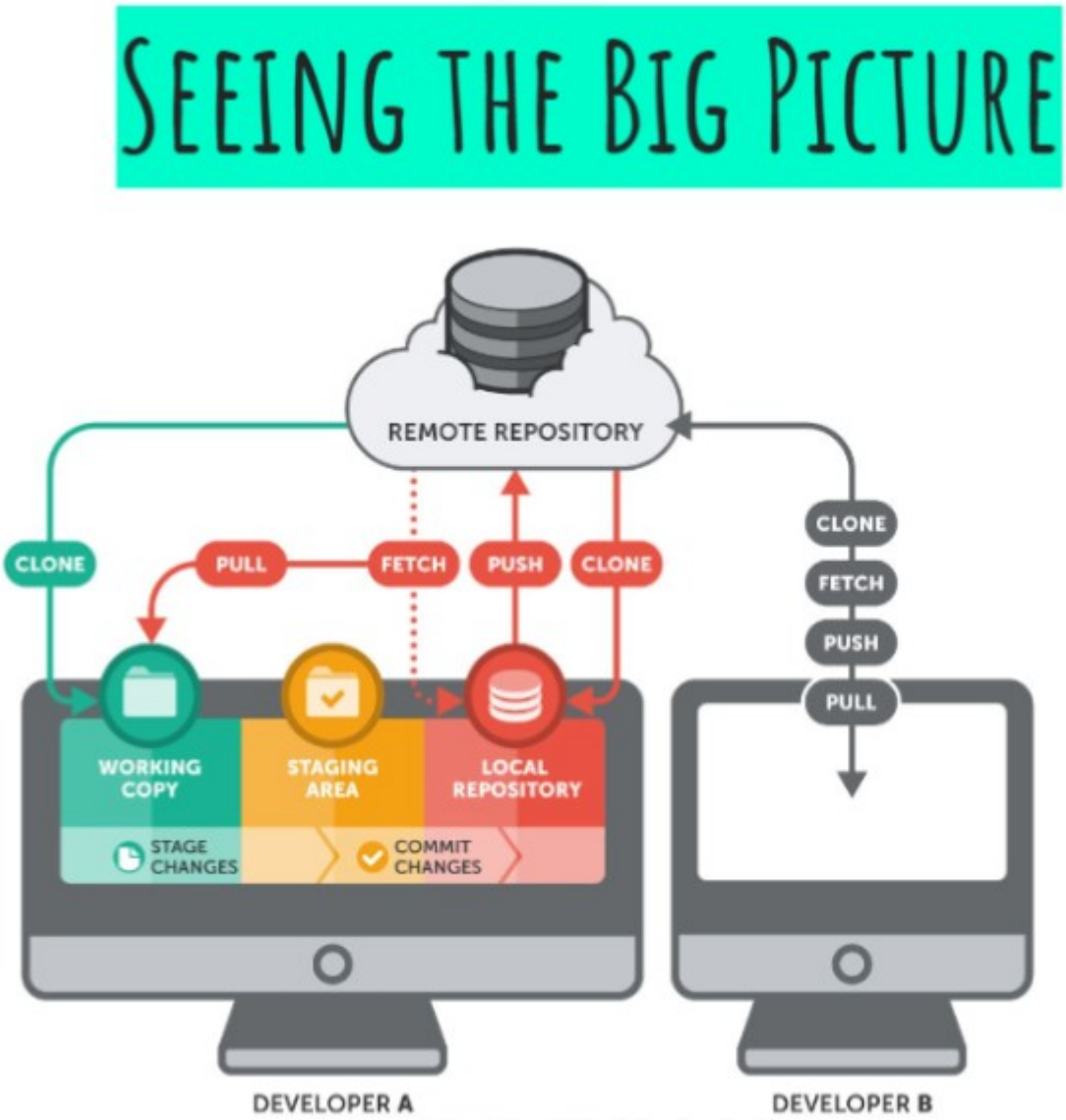
Deleting a Branch

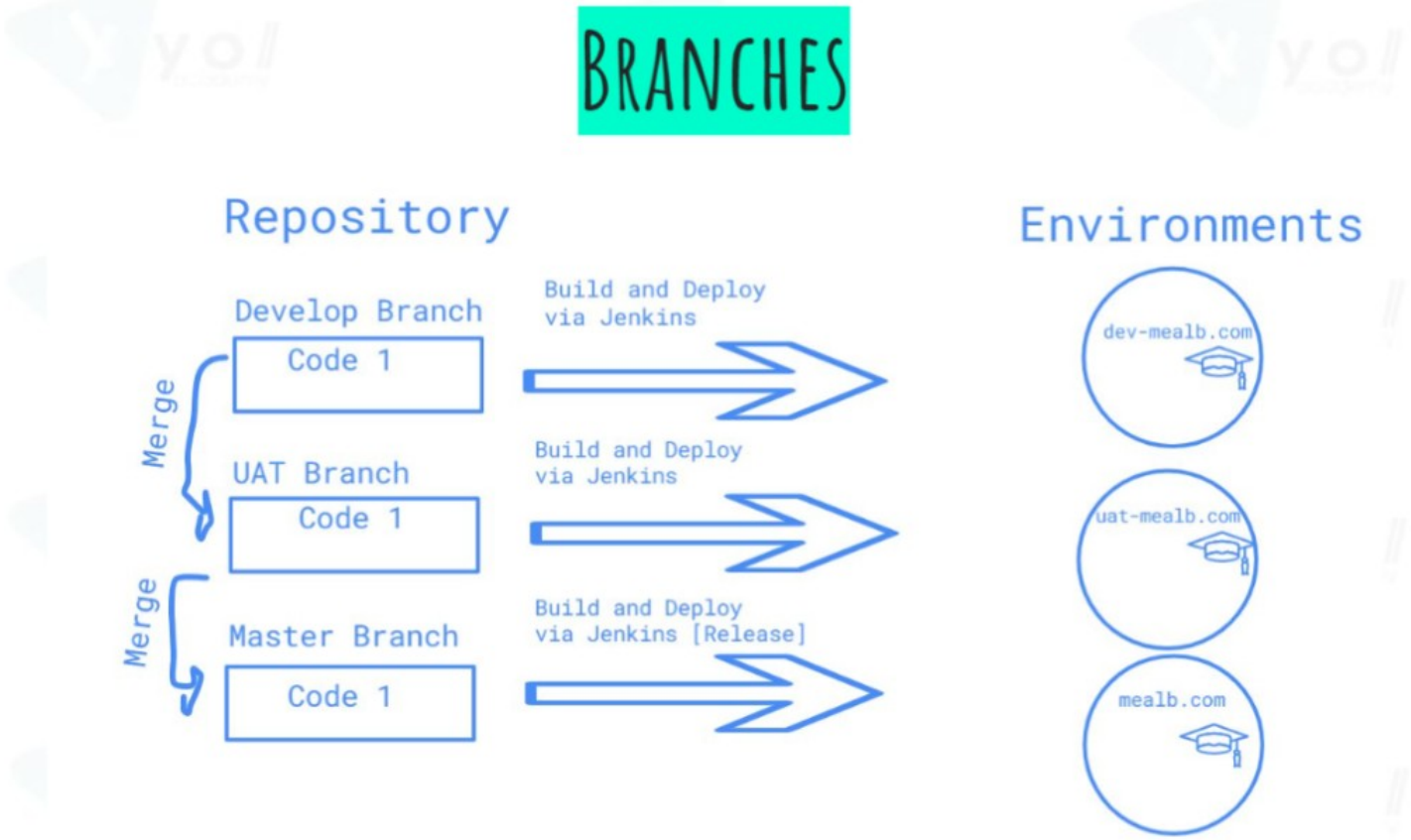
```
sh
CopyEdit
git branch -d feature-login
```

 Deletes the branch after merging (use `-D` if not merged).

Why Use Branches?

✔ Work on features independently
✔ Avoid breaking the main code
✔ Collaborate with teams more efficiently





The image visually represents how **branches, repositories, and environments** work together in a software development workflow. Here's a breakdown of the key elements:

1 Repository & Branches

- The **repository** contains different branches where code is managed.
- There are **three branches** in the repository:
  - **Develop Branch**: This is where active development happens. New features and bug fixes are added here.
  - **UAT (User Acceptance Testing) Branch**: After development, the code is merged into this branch for testing by users or QA teams.
  - **Master Branch**: This is the final, stable version of the code that gets deployed to production.
- **Merging Process**:
  - The **Develop Branch** merges into the **UAT Branch** after development.
  - The **UAT Branch** merges into the **Master Branch** when testing is complete.

2 Deployment Process

- **Each branch is built and deployed using Jenkins**:

- **Develop Branch** → Deployed to **DEV Environment** via Jenkins.
- **UAT Branch** → Deployed to **UAT Environment** via Jenkins.
- **Master Branch** → Deployed to **Production Environment** via Jenkins (as a Release).

3 Environments

- The code moves through **three different environments**:
  - **dev-mealb.com** → Development environment, where initial testing happens.
  - **uat-mealb.com** → UAT environment, where users or testers validate the changes.
  - **mealb.com** → Production environment, where the final version is live for real users.

Summary

This setup follows a **standard CI/CD pipeline**:

1. Developers work on the **Develop Branch**.
2. Once the feature is complete, it is **merged into UAT for testing**.
3. After testing, it is **merged into the Master Branch for release**.
4. Jenkins automates the **build and deployment** for each stage.

Understanding Branches, Environments, and Repositories in Git & Development

These three concepts work together to organize and manage software development. Let's break them down:

Branches (Feature Isolation & Parallel Development)

A **branch** in Git is an independent line of development that allows you to work on new features, bug fixes, or experiments **without affecting the main project**.

Main Branches:

- `main` (or `master`): The stable, production-ready version of the code.
- `develop`: The branch where active development happens before merging into `main`.

Feature Branches:

- `feature/login-page`: Created for a specific feature, then merged back into `develop`.
- `bugfix/payment-issue`: Created for fixing a bug, merged after testing.

🔹 Branch Lifecycle:
1 Create a branch → 2 Work on it → 3 Test → 4 Merge into `main` → 5 Delete if needed

Example Git Commands:

```
sh
CopyEdit
git branch feature-new-ui   # Create a branch
git switch feature-new-ui   # Move to the new branch
git commit -am "Added new UI"  # Commit changes
git merge feature-new-ui    # Merge into the main branch
git branch -d feature-new-ui  # Delete the branch
```

🌍 Environments (Where the Code Runs)

An **environment** is a stage where the application is deployed and tested before going live.

🔹 Common Environments:
1 **Development (`DEV`)** – Developers test their code here.
2 **Testing (`QA`)** – Quality assurance tests for bugs.
3 **Staging (`STG`)** – A pre-production environment, mimicking real-world use.
4 **Production (`PROD`)** – The live version of the application used by real users.

🔹 How Branches Relate to Environments:

- `develop` → **DEV**
- `release` → **STAGING**
- `main` → **PRODUCTION**

Example Workflow:
1 Create a `feature/new-checkout` branch
2 Merge into `develop` (DEV environment)
3 QA team tests it → Merges into `release` (STAGING)
4 If all tests pass, merge into `main` (PRODUCTION)

📦 Repository (Project Storage)

A **repository (repo)** is where your project's files, history, and branches are stored.

🔹 Types of Repositories:
1 **Local Repository** – Your copy on your computer.
2 **Remote Repository** – Stored on GitHub, GitLab, or Bitbucket.
3 **Forked Repository** – A copy of someone else's repo that you can modify.

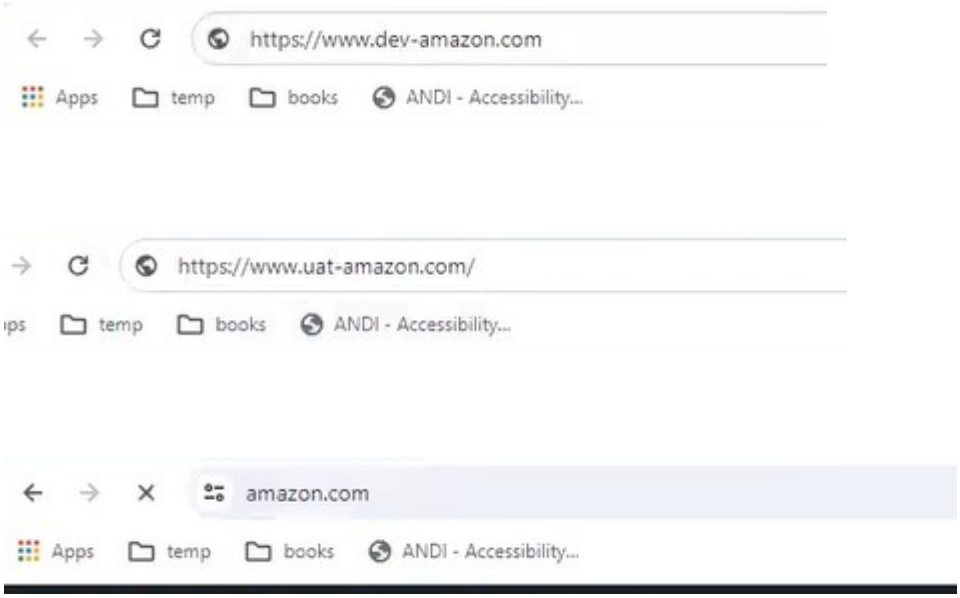
GitHub Example:

```
sh
CopyEdit
git clone https://github.com/user/project.git  # Copy a repo
git push origin feature-new-ui  # Push changes to GitHub
```

How They Work Together

1 **Developers create a branch** (e.g., `feature/new-ui`) inside a **repository**.
2 They work on it in their **local environment** and commit changes.
3 They push it to the **remote repository** for collaboration.
4 QA tests it in different **environments** (DEV → STAGING → PROD).
5 Once approved, it's merged into `main`, and users see the changes in **PRODUCTION**.

Example for Amazon:



BRANCHES

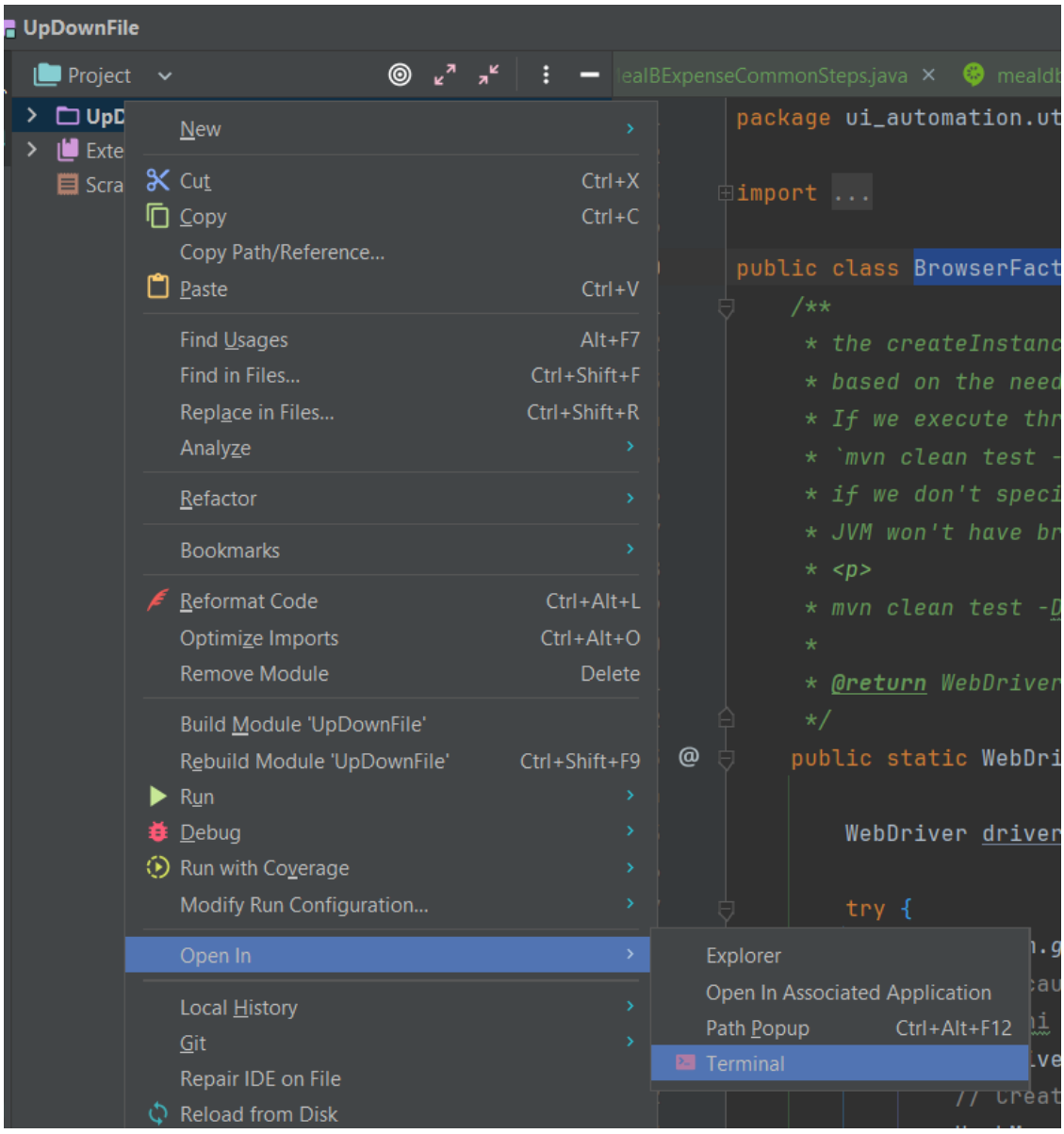
- All the changes you make at any time will **only apply to the currently active branch**; all other branches are left untouched.
- This gives you the freedom to both **work on different things in parallel** without worrying to mess up code in your master branch!

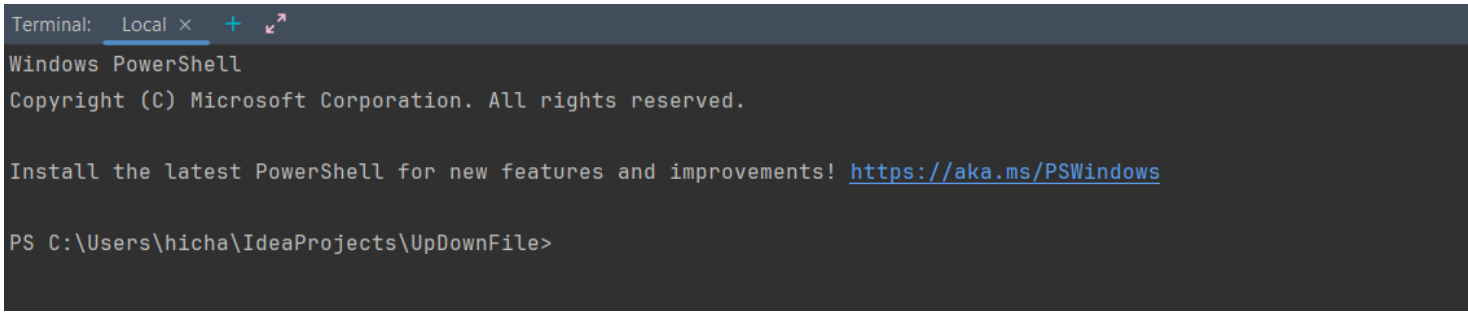
Commands:

- `git branch -a` → list all branches (remote and local)
- `git checkout -b <branchname>` → Create a branch and switch to it.
- `git checkout <branchname>` → Switch to a branch.
- `git branch -d <branchname>` → delete branch.
- `git push origin <branchname>` → Push the branch to your remote repository, so others can use it:

How to open the Terminal ?

Click right on the project /Open In / Terminal


UpDownFile
Project
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path/Reference...
Paste Ctrl+V
Find Usages Alt+F7
Find in Files... Ctrl+Shift+F
Replace in Files... Ctrl+Shift+R
Analyze
Refactor
Bookmarks
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Remove Module Delete
Build Module 'UpDownFile'
Rebuild Module 'UpDownFile' Ctrl+Shift+F9
Run
Debug
Run with Coverage
Modify Run Configuration...
Open In
Local History
Git
Repair IDE on File
Reload from Disk
Explorer
Open In Associated Application
Path Popup Ctrl+Alt+F12
Terminal
package ui_automation.ut
import ...
public class BrowserFact
* the createInstanc
* based on the need
* If we execute thr
* `mvn clean test -
* if we don't speci
* JVM won't have br
* <p>
* mvn clean test -D
* @return WebDriver
public static WebDri
WebDriver driver
try {


Terminal: Local
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\Users\hicha\IdeaProjects\UpDownFile>

getting started

start a new repo:

```
git init
```

clone an existing repo:

```
git clone $URL
```

know where you are

```
git status
```

prepare to commit

add untracked file:
(or unstaged changes)

```
git add $FILE
```

add ALL untracked files and unstaged changes:

```
git add .
```

choose which parts of a file to stage:

```
git add -p
```

delete or move file:

```
git rm $FILE
git mv $OLD $NEW
```

tell git to forget about a file without deleting it:

```
git rm --cached $FILE
```

unstage everything:

```
git reset HEAD
```

make commits

make a commit:
(and open a text editor to write the message)

```
git commit
```

make a commit:

```
git commit -m 'message'
```

commit all unstaged changes:

```
git commit -am 'message'
```

move between branches

switch branches:

```
git switch $NAME  OR
git checkout $NAME
```

create a branch:

```
git switch -c $NAME OR
git checkout -b $NAME
```

list branches:

```
git branch
```

delete a branch

```
git branch -d $NAME
```

force delete a branch:

```
git branch -D $NAME
```

list branches by most recently committed to:

```
git branch
  --sort=-committerdate
```

look at a branch's history

log the branch

```
git log main
```

show how two branches relate to each other:

```
git log --graph a b
```

one line log:

```
git log --oneline
```

code archaeology

show who last changed each line of a file:

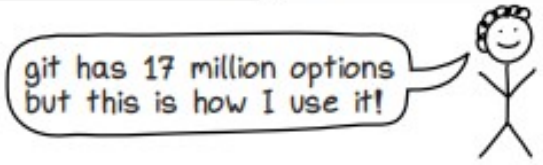
```
git blame $FILENAME
```

show every commit that modified a file:

```
git log $FILENAME
```

find every commit that added or removed some text:

```
git log -S banana
```



diff commits

show diff between a commit and its parent:

```
git show $COMMIT_ID
```

show diff between a merge commit and its merged parents:

```
git show --remerge-diff
        $COMMIT_ID
```

diff two commits:

```
git diff $COMMIT_ID $COMMIT_ID
```

just show diff for one file:

```
git diff $COMMIT_ID $FILENAME
```

show a summary of a diff:

```
git diff $COMMIT_ID --stat
git show $COMMIT_ID --stat
```

diff staged/unstaged changes

diff all staged and unstaged changes:

```
git diff HEAD
```

diff just staged changes:

```
git diff --staged
```

diff just unstaged changes:

```
git diff
```

configure git

set a config option:

```
git config user.name 'Julia'
           name      value
```

see all possible config options:

```
man git-config
```

set option globally:

```
git config --global ...
```

add an alias:

```
git config alias.st status
```

important git files

local git config:

```
.git/config
```

global git config:

```
~/.gitconfig
```

list of files to ignore:

```
.gitignore
```

trash your changes

delete all staged and unstaged changes to one file:

```
git checkout HEAD $FILE
```

delete unstaged changes to one file:

```
git checkout $FILE
```

delete all staged and unstaged changes:

```
git reset --hard
```

delete untracked files:

```
git clean
```

"stash" all staged and unstaged changes (pretend I might get them back later)

```
git stash
```

edit history

"undo" the most recent commit (keep your working directory the same):

```
git reset HEAD^
```

squash the last 5 commits into one:

```
git rebase -i HEAD^^^^^^
```

(and change "pick" to "fixup" for any commit I want to combine with the previous one)

undo a failed rebase:

①`git reflog BRANCHNAME`
②do a painstaking search
③`git reset --hard $COMMIT_ID`

change a commit message: (or add a file you forgot)

```
git commit --amend
```

restore an old file

get the version of a file from another branch or commit

```
git checkout $COMMIT_ID $FILE
```

OR

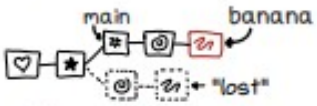
```
git restore $FILE
    --source $COMMIT_ID
```

combine diverged branches

how the branches look before:

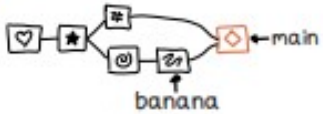
→ combine with rebase:

```
git switch banana
git rebase main
```



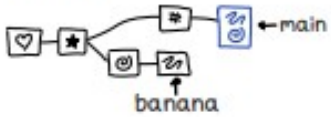
→ combine with merge:

```
git switch main
git merge banana
git commit
```



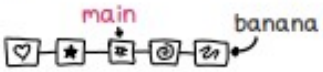
→ combine with squash merge:

```
git switch main
git merge --squash banana
git commit
```

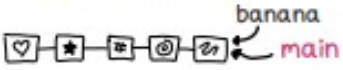


bring a branch up to date with another branch

(aka "fast-forward merge")




```
git switch main
git merge banana
```



copy one commit onto another branch

before:

```
git cherry-pick $COMMIT_ID
```

after:

add a remote

```
git remote add $NAME $URL
```

push your changes

push the `main` branch to the remote `origin`:

```
git push origin main
```

push a branch to the remote `origin` that you've never pushed before:

```
git push -u origin $NAME
```

push the current branch to its remote "tracking branch":

```
git push
```

force push: ⚠

```
git push --force-with-lease
```

push tags:

```
git push --tags
```

pull changes

fetch changes:

(but don't change any of your local branches)

```
git fetch origin main
```

fetch changes and then merge them into your current branch:

```
git pull origin main
```

OR

```
git pull
```

fetch changes and then rebase your current branch:

```
git pull --rebase
```

fetch all branches:

```
git fetch --all
```

ways to refer to a commit

every time we say $COMMIT_ID, you can use any of these:

a branch	`main`
a tag	`v0.1`
a commit ID	`3e887ab`
a remote branch	`origin/main`
current commit	`HEAD`
3 commits ago	`HEAD^^^`
3 commits ago	`HEAD~3`

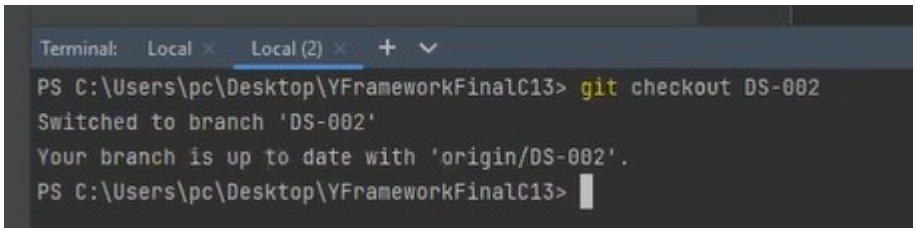
- **List all branches:**

```
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git branch -a
* DS-001
  DS-002
  master
  remotes/origin/DS-001
  remotes/origin/DS-002
  remotes/origin/HEAD -> origin/master
  remotes/origin/dev
  remotes/origin/master
```

The local branches are DS-001, DS-002 and master

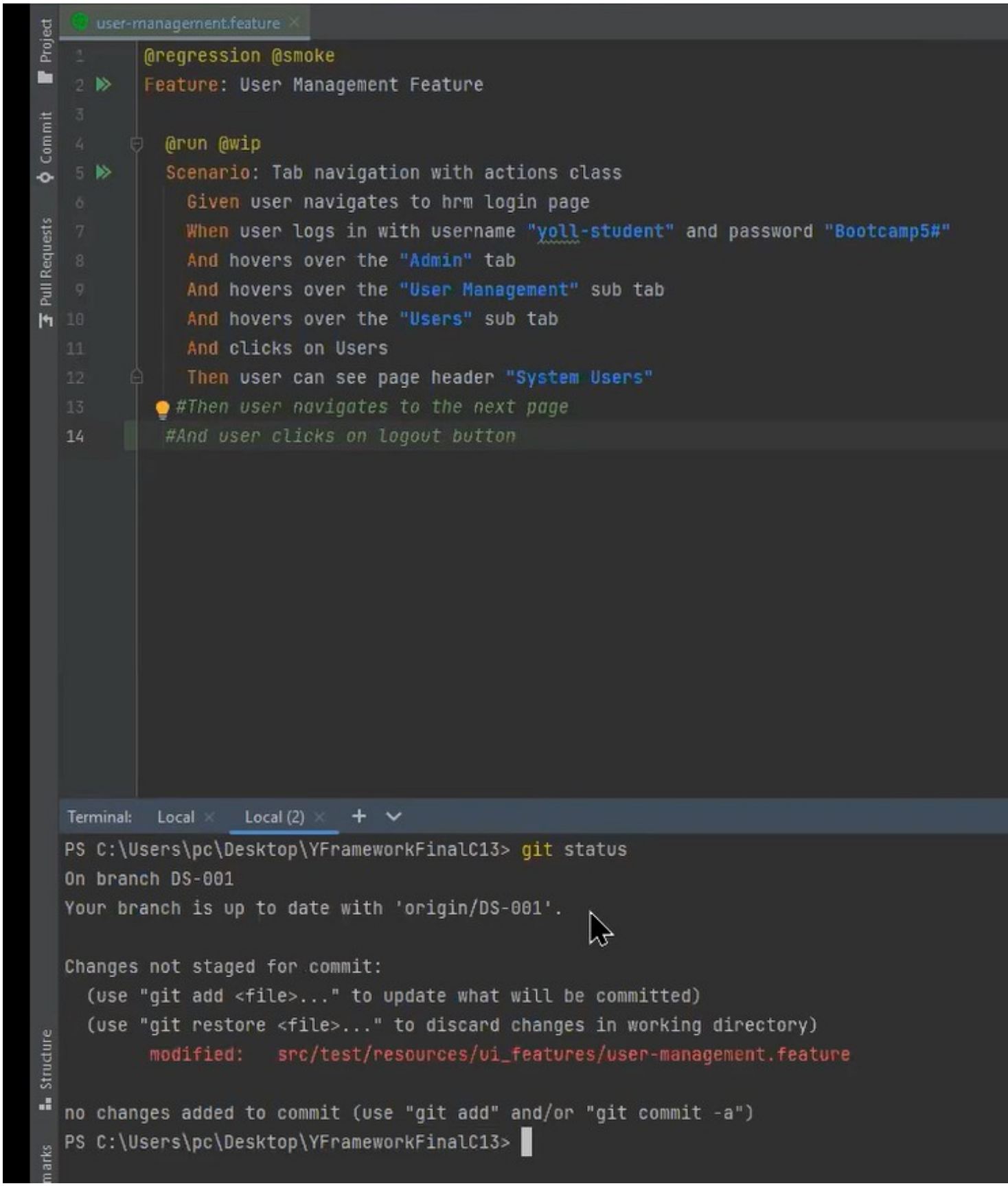
The branches from Repository are on RED Color

- **To Check out DS-002**


Terminal: Local Local (2)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git checkout DS-002
Switched to branch 'DS-002'
Your branch is up to date with 'origin/DS-002'.
PS C:\Users\pc\Desktop\YFrameworkFinalC13>

- **Get the status:**

It tells me which file I did modification

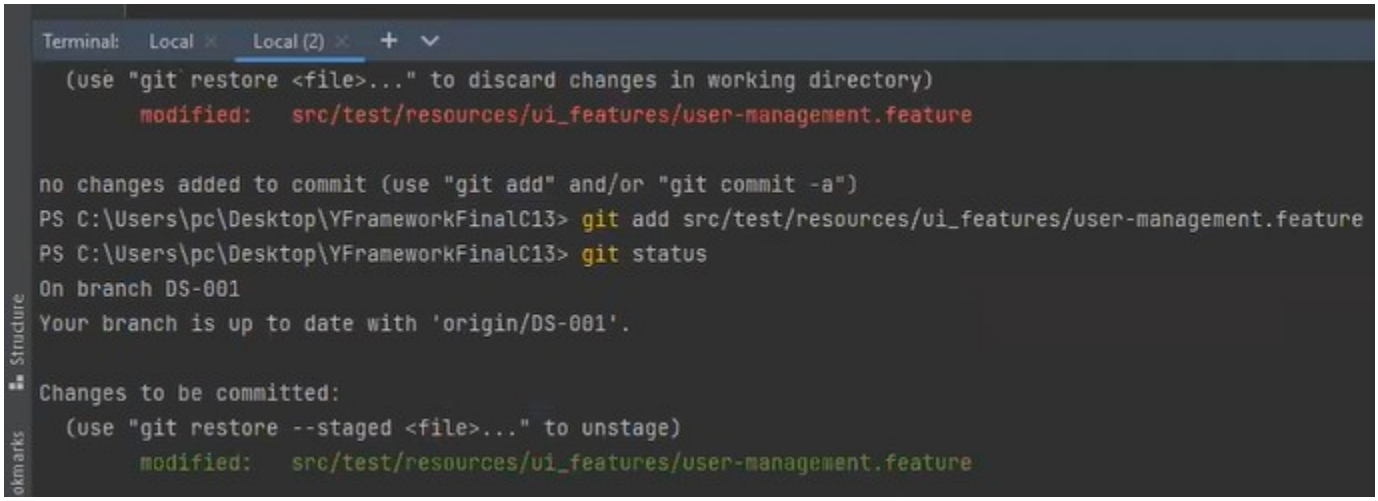

user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on logout button
Terminal: Local Local (2)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git status
On branch DS-001
Your branch is up to date with 'origin/DS-001'.
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified: src/test/resources/ui_features/user-management.feature
no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\pc\Desktop\YFrameworkFinalC13>

- **Add to stage Area**

```
no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git add src/test/resources/ui_features/user-management.feature
```

Before, it was red color which means the file is not in the stage area.

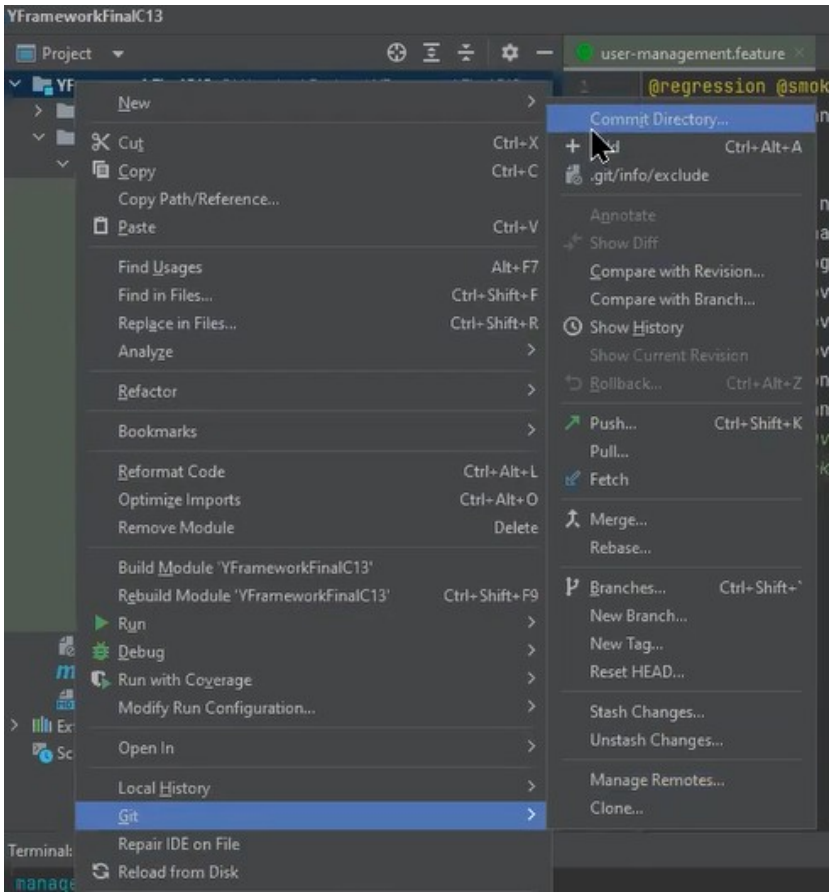
After Adding it, we can see the green color which means the file is in the stage area.


Terminal: Local Local (2)
(use "git restore <file>..." to discard changes in working directory)
modified: src/test/resources/ui_features/user-management.feature
no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git add src/test/resources/ui_features/user-management.feature
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git status
On branch DS-001
Your branch is up to date with 'origin/DS-001'.
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
modified: src/test/resources/ui_features/user-management.feature
Structure
okmarks

- **Add a message :**

PS C:\Users\pc\Desktop\YFrameworkFinalC13> git commit -m "I updated user management feature file"
[DS-001 58ca175] I updated user management feature file
1 file changed, 2 insertions(+), 1 deletion(-)

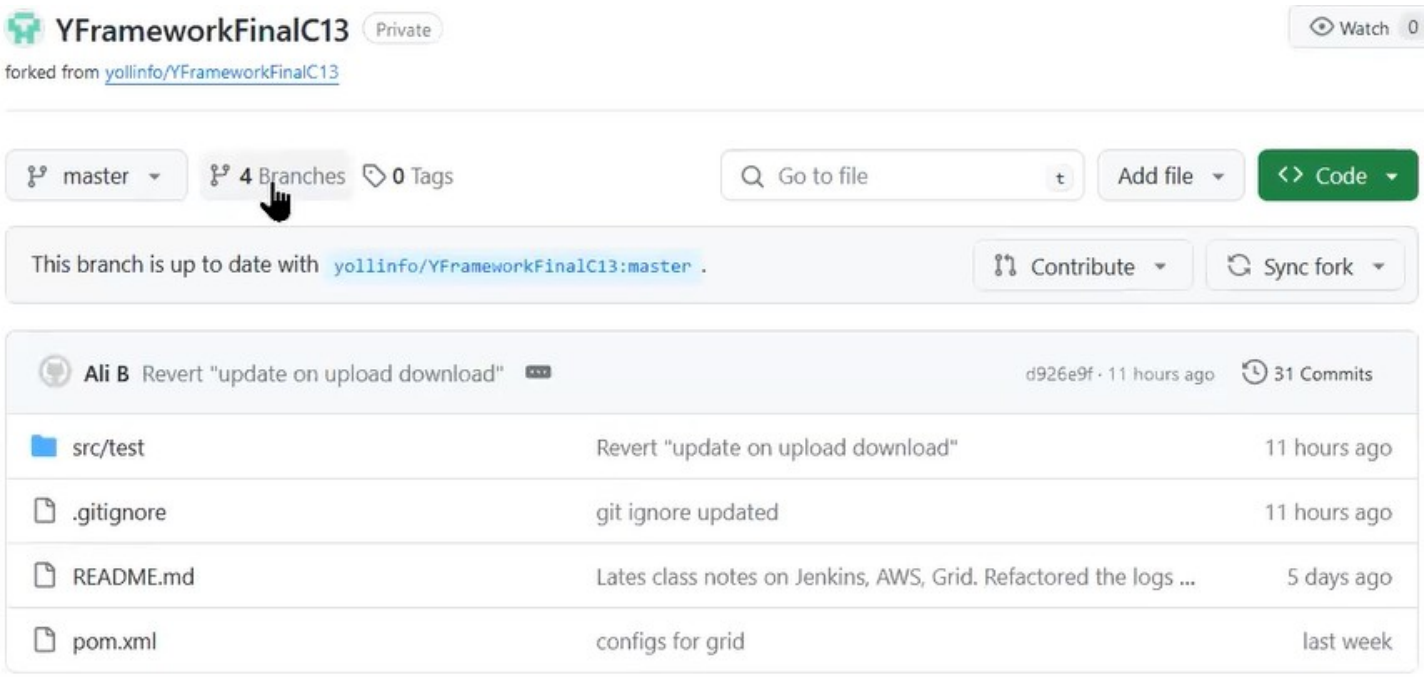
How to see the Comment ? Click right on the project/ Git/ Comment Directory

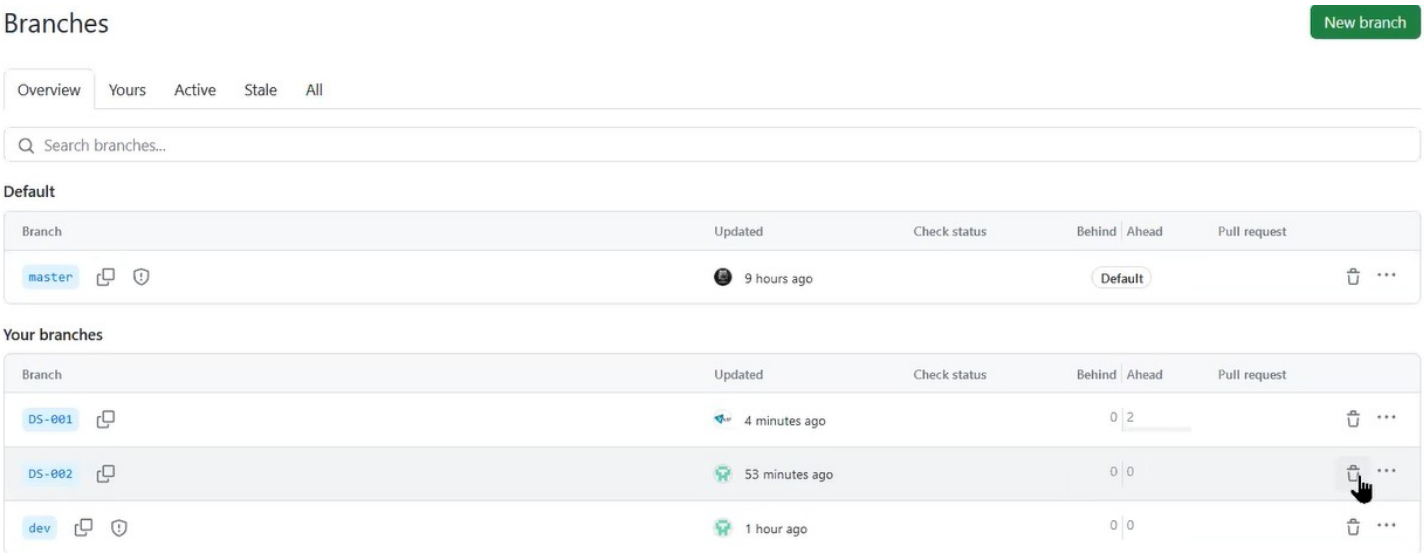


- **Push :**

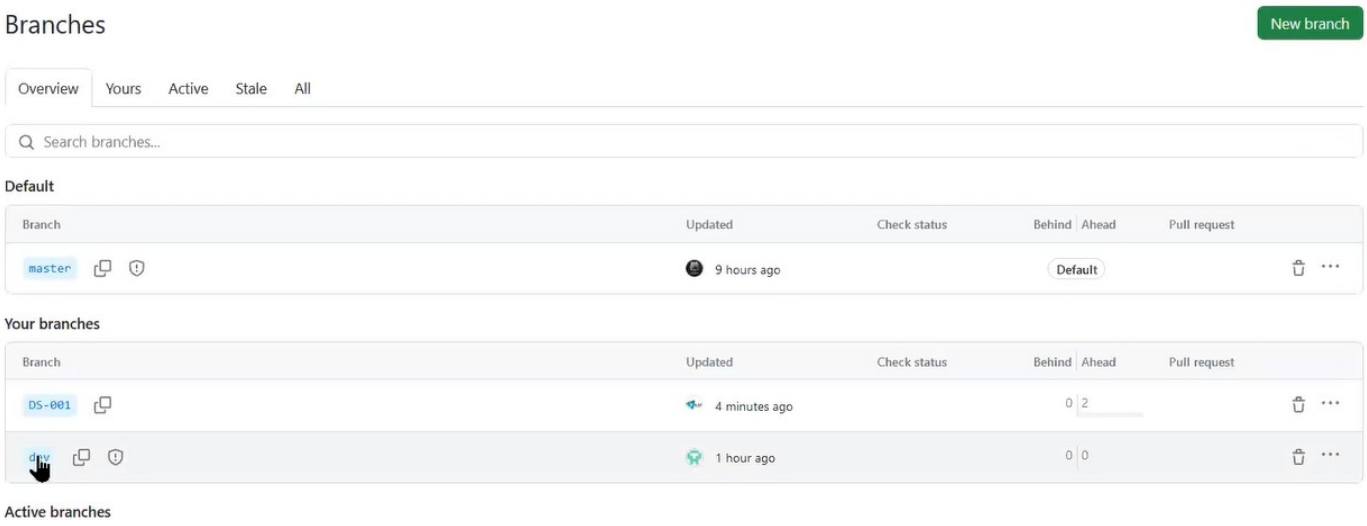
```
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git push
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 571 bytes | 571.00 KiB/s, done.
Total 7 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/AliBoozkurt/YFrameworkFinalC13.git
   a8dd048..58ca175  DS-001 -> DS-001
```

- **Delete a branch:**
- **Go to Branches / Delete DS-002**

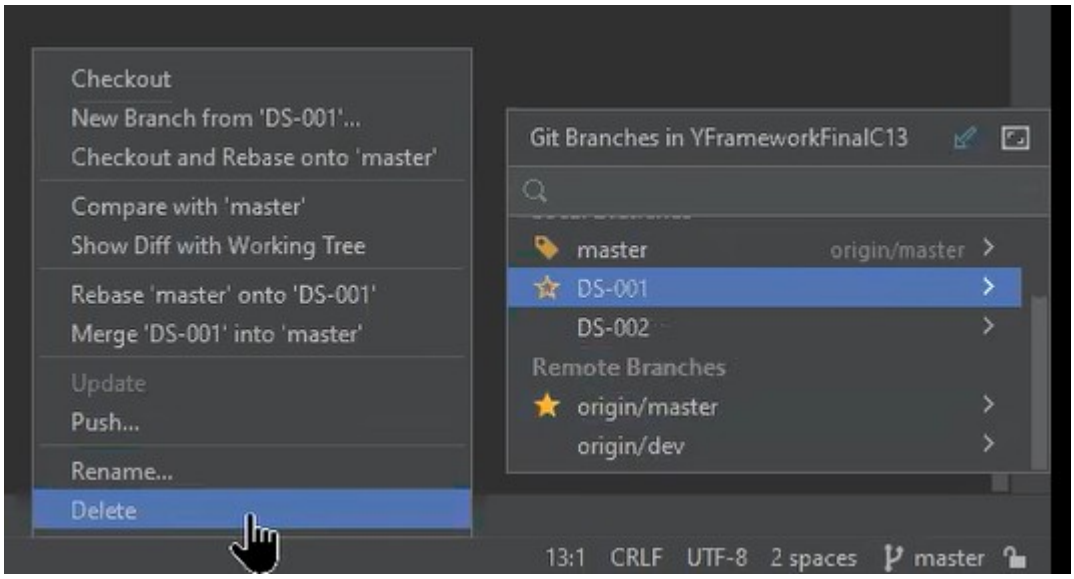

YFrameworkFinalC13
Private
forked from yollinfo/YFrameworkFinalC13
Watch 0
master
4 Branches
0 Tags
Go to file
t
Add file
Code
This branch is up to date with yollinfo/YFrameworkFinalC13:master.
Contribute
Sync fork
Ali B Revert "update on upload download"
d926e9f · 11 hours ago
31 Commits
src/test
Revert "update on upload download"
11 hours ago
.gitignore
git ignore updated
11 hours ago
README.md
Lates class notes on Jenkins, AWS, Grid. Refactored the logs ...
5 days ago
pom.xml
configs for grid
last week


Branches
New branch
Overview
Yours
Active
Stale
All
Search branches...
Default
Branch
Updated
Check status
Behind
Ahead
Pull request
master
9 hours ago
Default
Your branches
DS-001
4 minutes ago
0 2
DS-002
53 minutes ago
0 0
dev
1 hour ago
0 0

Refresh


Branches
New branch
Overview
Yours
Active
Stale
All
Search branches...
Default
Branch
Updated
Check status
Behind
Ahead
Pull request
master
9 hours ago
Default
Your branches
DS-001
4 minutes ago
0 2
dev
1 hour ago
0 0
Active branches

We can delete a file MANUALLY FROM INTELLIJ



The image presents a comparison between two repository structures for managing development and testing code:

Title:

"Dev and Test Repos" (highlighted in a cyan-colored box)

Option 1: Separate Repositories

- **Development Repo**
  - Contains branches
  - Has a `Master` branch where developers' code resides
  - Caution is required because production relies on this repo
- **Testing Repo (1.1 & 1.2)**
  - Contains branches
  - Has a `Master` branch
  - The second testing repo (1.2) includes additional branches:
    - `MB01`
    - `MB02`

Option 2: Combined Repository

- A single repository containing both **development** and **testing** code

Key Difference:

- **Option 1** separates development and testing into different repositories, ensuring isolation but requiring extra management.
- **Option 2** consolidates everything into one repository, simplifying management but possibly increasing complexity in maintaining different environments.

STASHING

- There are times when you have made changes to some code and don't want to commit.
- Like you write an email and not ready to send, so you **save it as draft.**
- Because your changes are not complete. In these scenarios, we **stash** those changes.
  - IntelliJ: VCS → GIT → Uncommitted changes → Stash Changes

- Think of the Stash as a draft email: it takes all the changes in your working copy and saves them for you. You're left with a clean working directory: you have no more local changes.
- Later, at any time, you can restore the changes from that clipboard in your working copy (**unstash**) - and continue working where you left off.
  - IntelliJ: VCS → GIT → Unstash Changes

Explanation of Stashing:

- Sometimes, when making changes in your code, you may not want to commit immediately.
- This is similar to writing an email but not sending it yet; instead, you **save it as a draft**.
- In Git, when your changes are not complete, you use **stash** to temporarily save them.

How Stashing Works:

1. **Saving Changes to Stash:**
   - Similar to storing an unfinished email as a draft.
   - Stashing removes changes from the working directory but keeps them safe.
   - In **IntelliJ IDEA**, you can stash changes via:
     VCS → GIT → Uncommitted changes → Stash Changes
2. **Retrieving Stashed Changes (Unstash):**
   - Later, when you are ready, you can retrieve the stashed changes.
   - This is like opening a draft email and continuing where you left off.
   - In **IntelliJ IDEA**, you can unstash using:
     VCS → GIT → Unstash Changes

Key Takeaways:

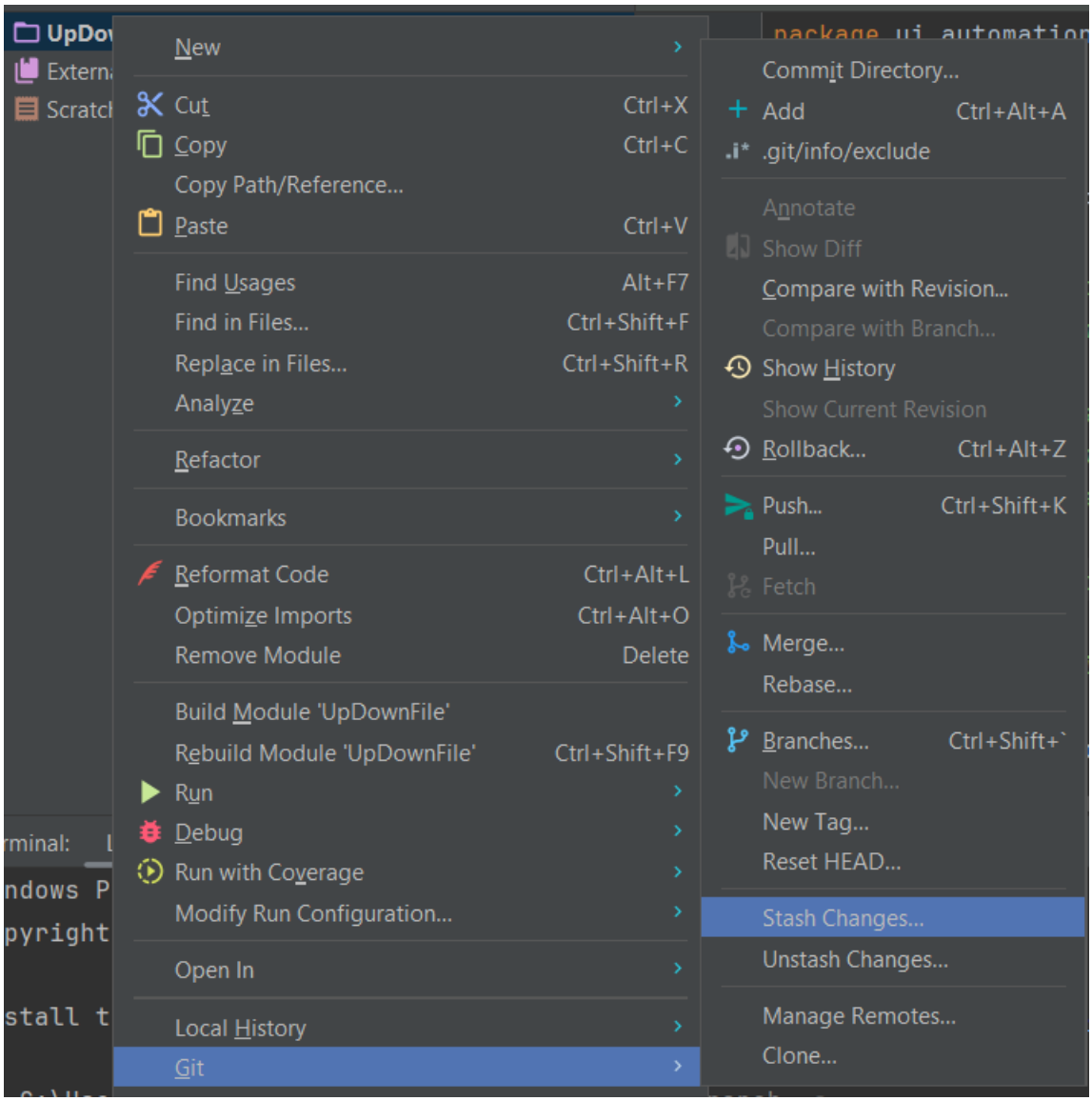
- **Stashing** is useful when you need to temporarily save changes but keep a clean working directory.
- You can **unstash** at any time to restore the saved changes and continue working.

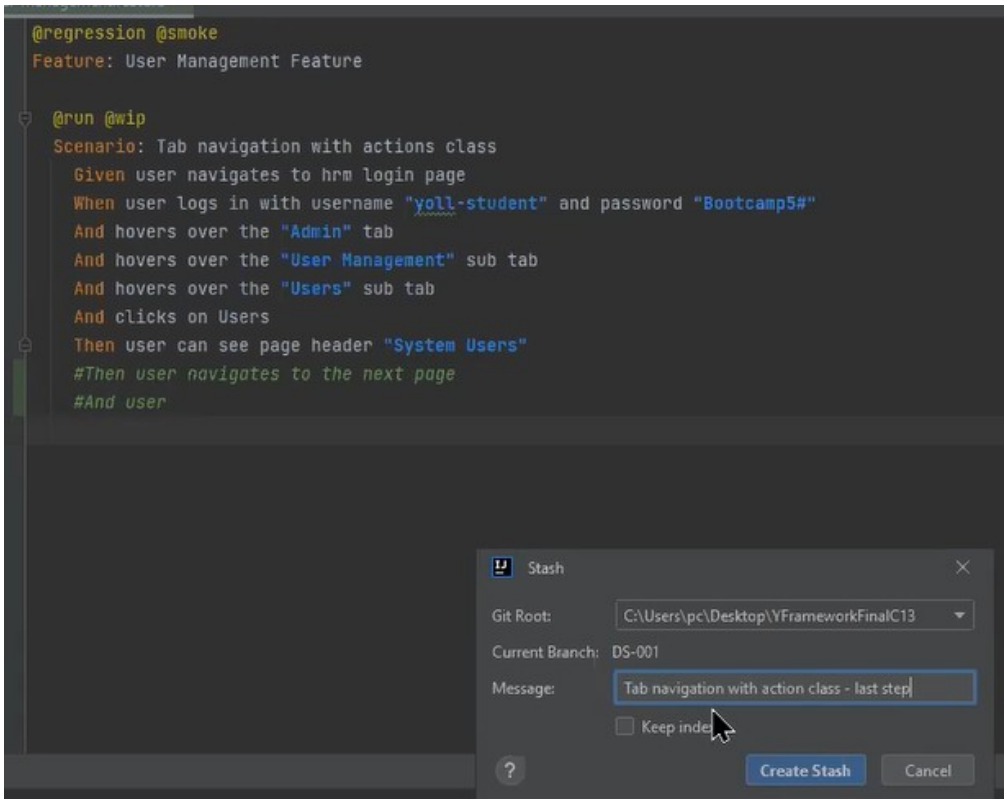
Example:

The tester wants to keep some changes without pushing them yet.

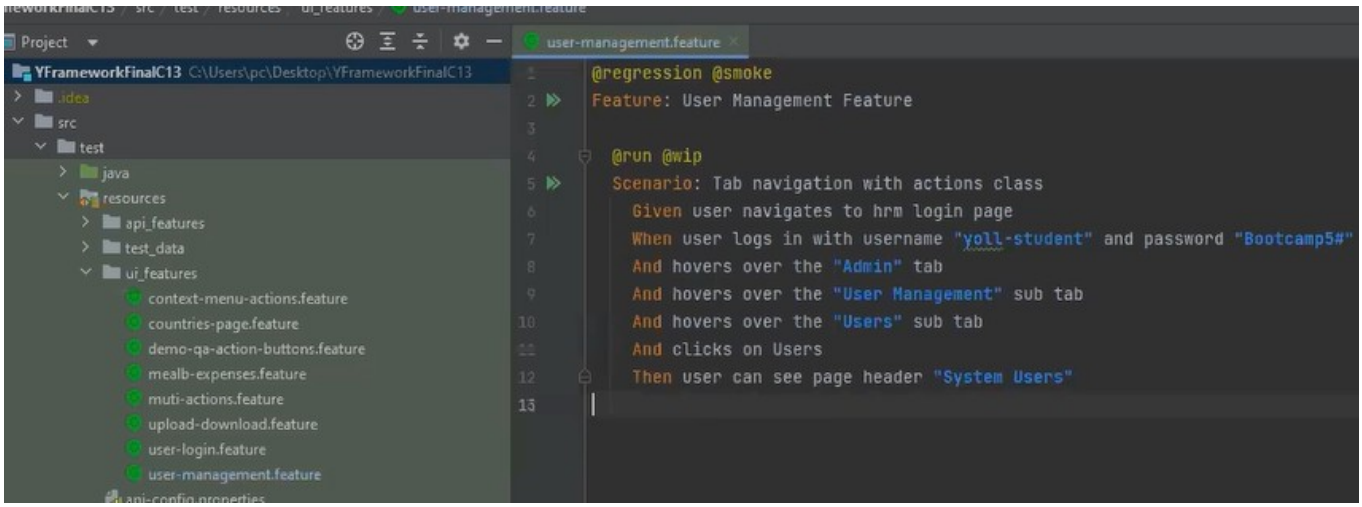
```
@regression @smoke
Feature: User Management Feature

  @run @wip
  Scenario: Tab navigation with actions class
    Given user navigates to hrm login page
    When user logs in with username "yoll-student" and password "Bootcamp5#"
    And hovers over the "Admin" tab
    And hovers over the "User Management" sub tab
    And hovers over the "Users" sub tab
    And clicks on Users
    Then user can see page header "System Users"
    #Then user navigates to the next page
    #And user
```


New
Cut Ctrl+X
Copy Ctrl+C
Copy Path/Reference...
Paste Ctrl+V
Find Usages Alt+F7
Find in Files... Ctrl+Shift+F
Replace in Files... Ctrl+Shift+R
Analyze
Refactor
Bookmarks
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Remove Module Delete
Build Module 'UpDownFile'
Rebuild Module 'UpDownFile' Ctrl+Shift+F9
Run
Debug
Run with Coverage
Modify Run Configuration...
Open In
Local History
Git
Commit Directory...
Add Ctrl+Alt+A
.git/info/exclude
Annotate
Show Diff
Compare with Revision...
Compare with Branch...
Show History
Show Current Revision
Rollback... Ctrl+Alt+Z
Push... Ctrl+Shift+K
Pull...
Fetch
Merge...
Rebase...
Branches... Ctrl+Shift+`
New Branch...
New Tag...
Reset HEAD...
Stash Changes...
Unstash Changes...
Manage Remotes...
Clone...


@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user
Stash
Git Root:
C:\Users\pc\Desktop\YFrameworkFinalC13
Current Branch:
DS-001
Message:
Tab navigation with action class - last step
Keep inde
Create Stash
Cancel

After clicking on Create Stach button, the changes will be gone in the memory without seeing it.

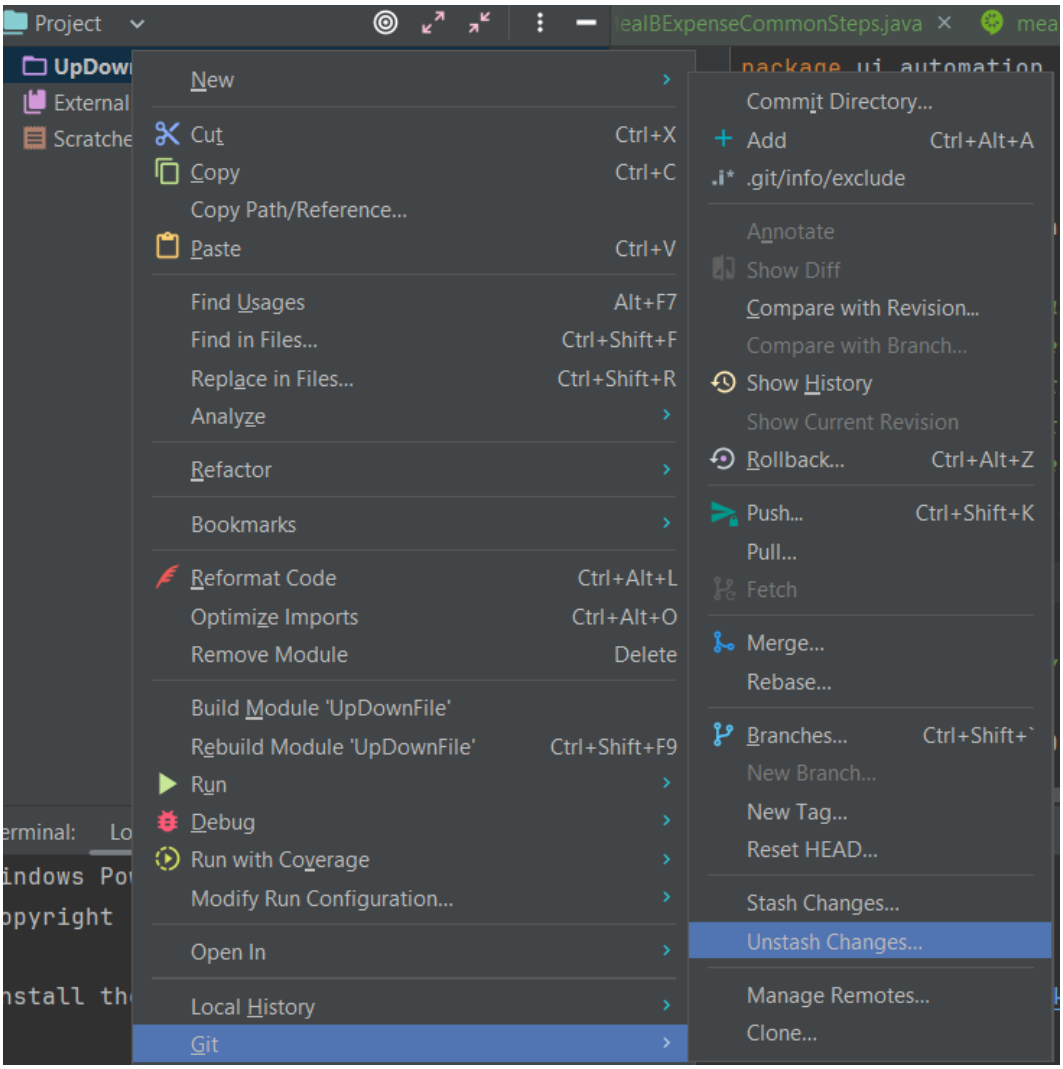


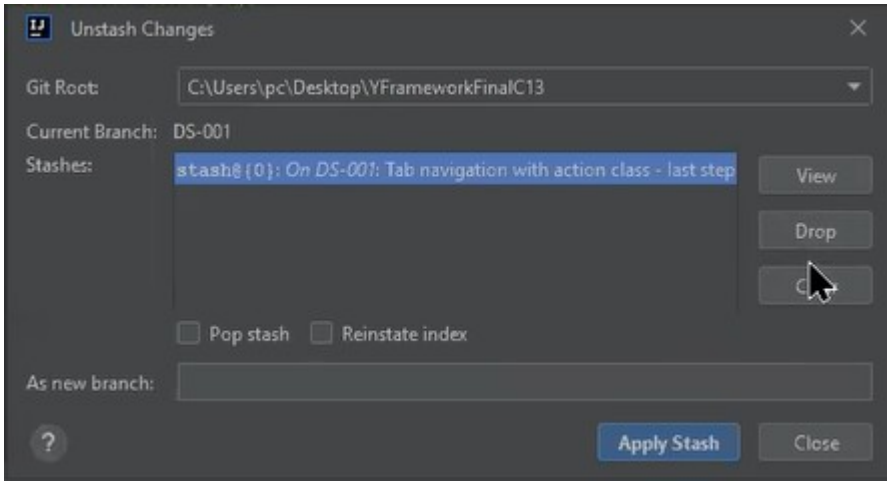
How to see again the changes after they were disappeared ?

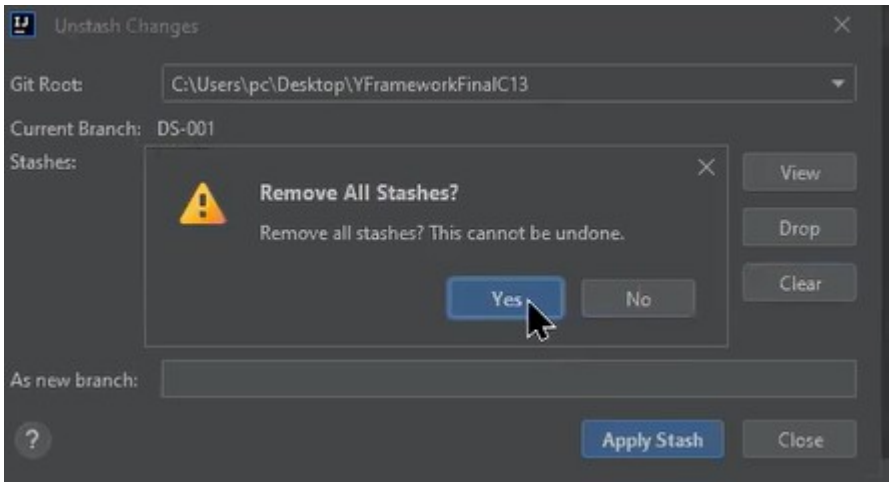
Choose the banch

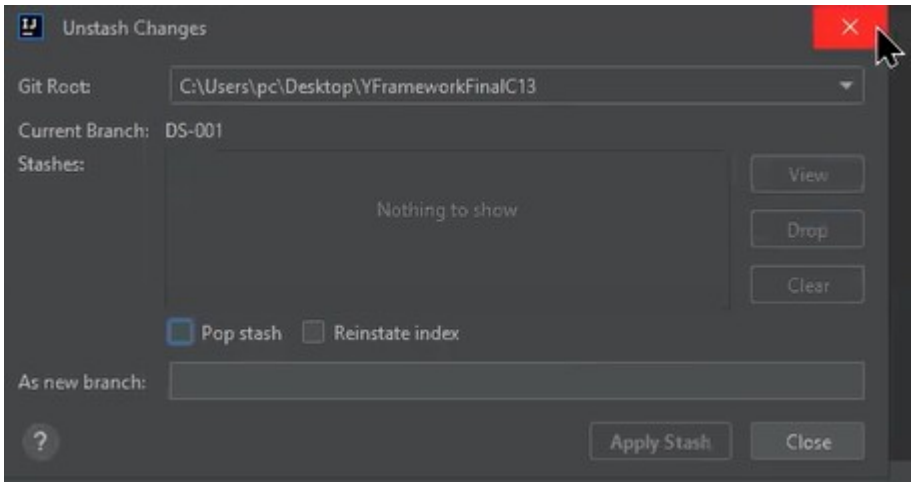
- In **IntelliJ IDEA**, you can unstash using:
  GIT / Unstash Changes
- Click on **Clear** / **Remove All Stashes** / close

o


Project
UpDow
External
Scratche
ealBExpenseCommonSteps.java
mea
package ui_automation
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path/Reference...
Paste Ctrl+V
Find Usages Alt+F7
Find in Files... Ctrl+Shift+F
Replace in Files... Ctrl+Shift+R
Analyze
Refactor
Bookmarks
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Remove Module Delete
Build Module 'UpDownFile'
Rebuild Module 'UpDownFile' Ctrl+Shift+F9
Run
Debug
Run with Coverage
Modify Run Configuration...
Open In
Local History
Git
Commit Directory...
Add Ctrl+Alt+A
.git/info/exclude
Annotate
Show Diff
Compare with Revision...
Compare with Branch...
Show History
Show Current Revision
Rollback... Ctrl+Alt+Z
Push... Ctrl+Shift+K
Pull...
Fetch
Merge...
Rebase...
Branches... Ctrl+Shift+`
New Branch...
New Tag...
Reset HEAD...
Stash Changes...
Unstash Changes...
Manage Remotes...
Clone...
erminal:
indows Po
opyright
nstall th


Unstash Changes
Git Root: C:\Users\pc\Desktop\YFrameworkFinalC13
Current Branch: DS-001
Stashes:
stash@{0}: On DS-001: Tab navigation with action class - last step
View
Drop
Pop stash
Reinstate index
As new branch:
Apply Stash
Close

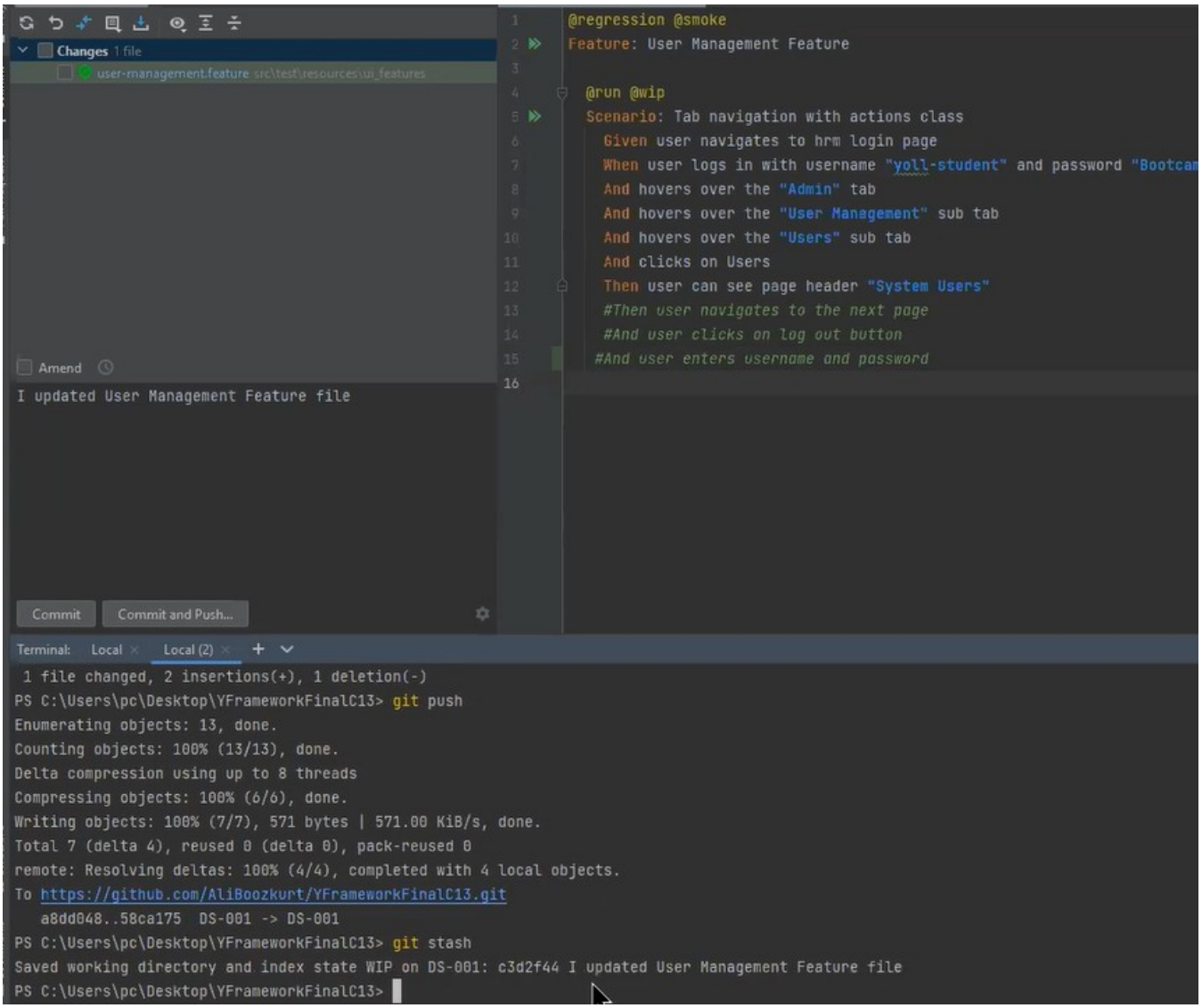




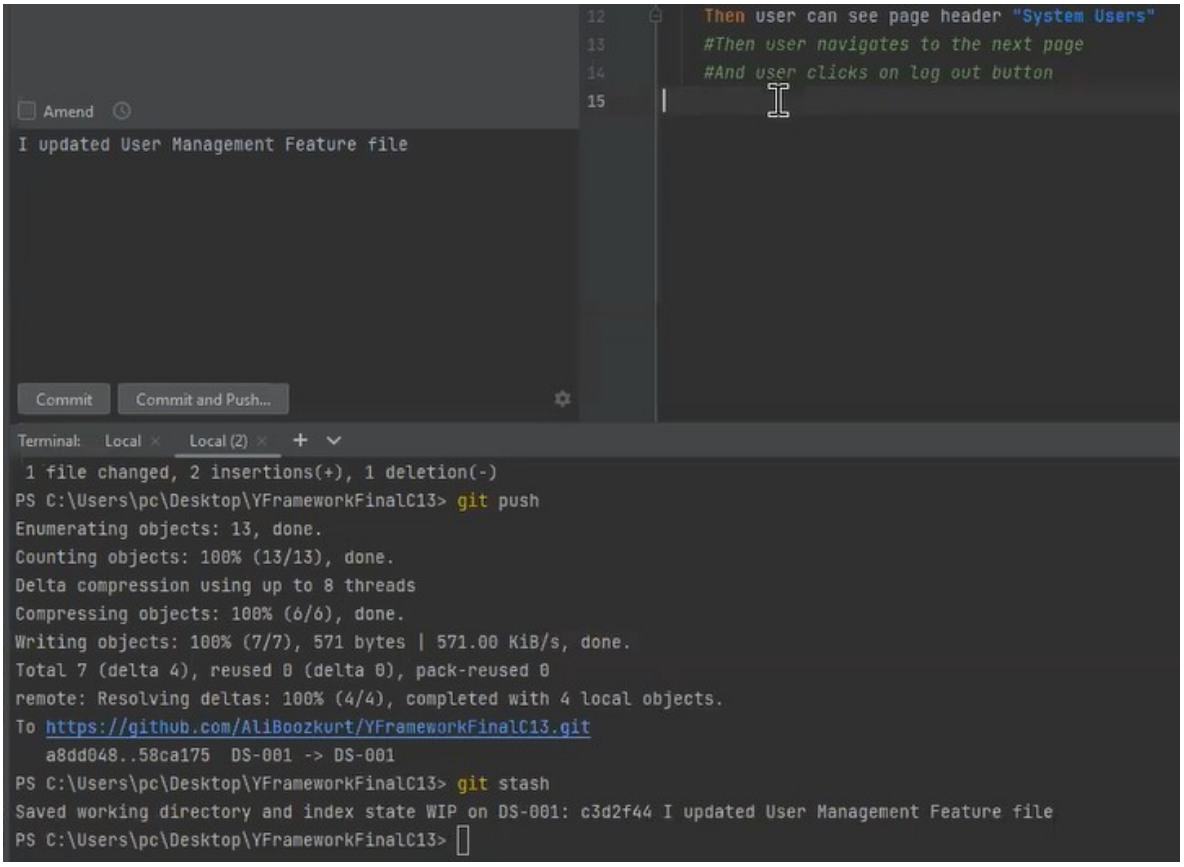
Adding another line

#And user enters username and password.

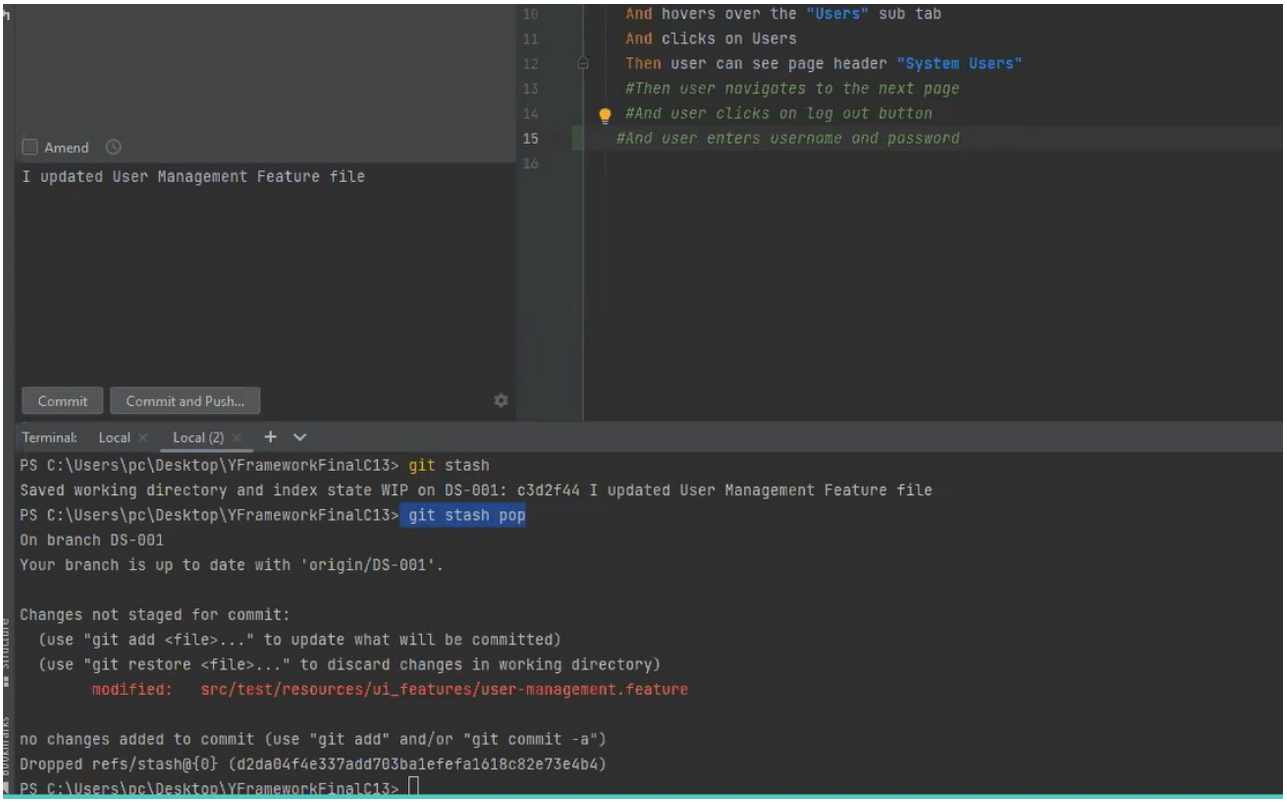
Use : git stash


Changes 1 file
user-management.feature src\test\resources\ui_features
Amend
I updated User Management Feature file
Commit
Commit and Push...
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcam
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
Terminal: Local Local (2)
1 file changed, 2 insertions(+), 1 deletion(-)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git push
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 571 bytes | 571.00 KiB/s, done.
Total 7 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/AliBoozkurt/YFrameworkFinalC13.git
a8dd048..58ca175 DS-001 -> DS-001
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git stash
Saved working directory and index state WIP on DS-001: c3d2f44 I updated User Management Feature file
PS C:\Users\pc\Desktop\YFrameworkFinalC13>

Then the third line is disappeared

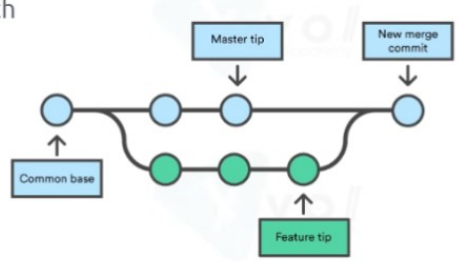

Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
Amend
I updated User Management Feature file
Commit
Commit and Push...
Terminal: Local Local (2)
1 file changed, 2 insertions(+), 1 deletion(-)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git push
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 571 bytes | 571.00 KiB/s, done.
Total 7 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/AliBoozkurt/YFrameworkFinalC13.git
a8dd048..58ca175 DS-001 -> DS-001
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git stash
Saved working directory and index state WIP on DS-001: c3d2f44 I updated User Management Feature file
PS C:\Users\pc\Desktop\YFrameworkFinalC13>

To make it appearing again, use : git stash pop


And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
Amend
I updated User Management Feature file
Commit
Commit and Push...
Terminal: Local Local (2)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git stash
Saved working directory and index state WIP on DS-001: c3d2f44 I updated User Management Feature file
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git stash pop
On branch DS-001
Your branch is up to date with 'origin/DS-001'.
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified: src/test/resources/ui_features/user-management.feature
no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (d2da04f4e337add703ba1efefa1618c82e73e4b4)
PS C:\Users\pc\Desktop\YFrameworkFinalC13>

MERGE

Merging → is the act of **integrating another branch into your current working branch.** You're taking changes from another branch and combine them with your current working branch.



- A great thing about having Git is that, it makes merging extremely easy: in most cases, Git will figure out how to integrate new changes.
- Git will figure out which commits you don't have in your current working branch. Only these commits will be integrated.

`git merge <branchname>` → To merge a different branch into your active branch.

What is `git merge`?

`git merge` is a Git command used to combine the changes from one branch into another. It integrates the changes from a source branch into the target branch, creating a new commit if necessary.

How `git merge` Works

1. **Fetch Changes (If Needed):**
   Before merging, ensure you have the latest updates by running:

   ```
   bash
   CopyEdit
   git fetch origin
   ```

2. **Switch to the Target Branch:**

   ```
   bash
   CopyEdit
   git checkout main  # or git switch main
   ```

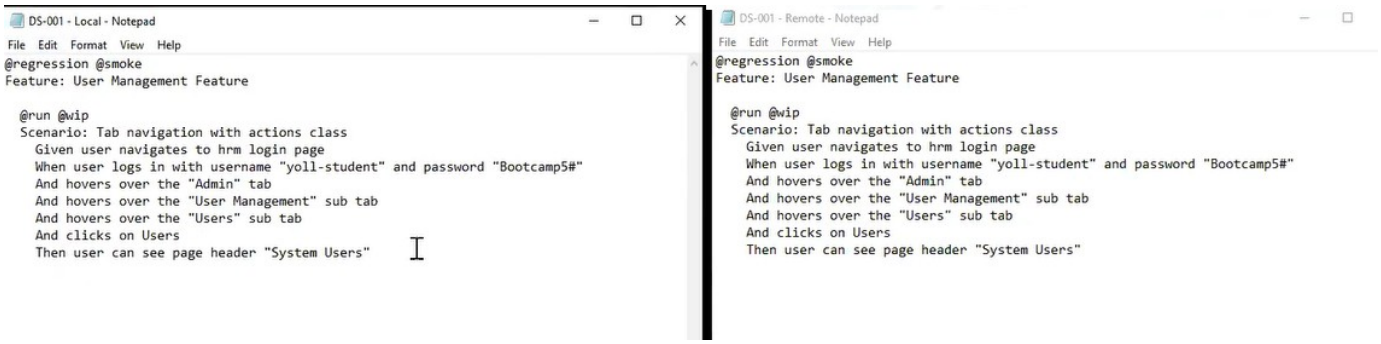
3. **Merge the Source Branch into the Target Branch:**

   ```
   bash
   CopyEdit
   git merge feature-branch
   ```

   This merges `feature-branch` into `main`.

Example to understand:

I have two branches: DS-001 Local, and DS-001 Remote


DS-001 - Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
DS-001 - Remote - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"

I did a change on the Local DS-001.

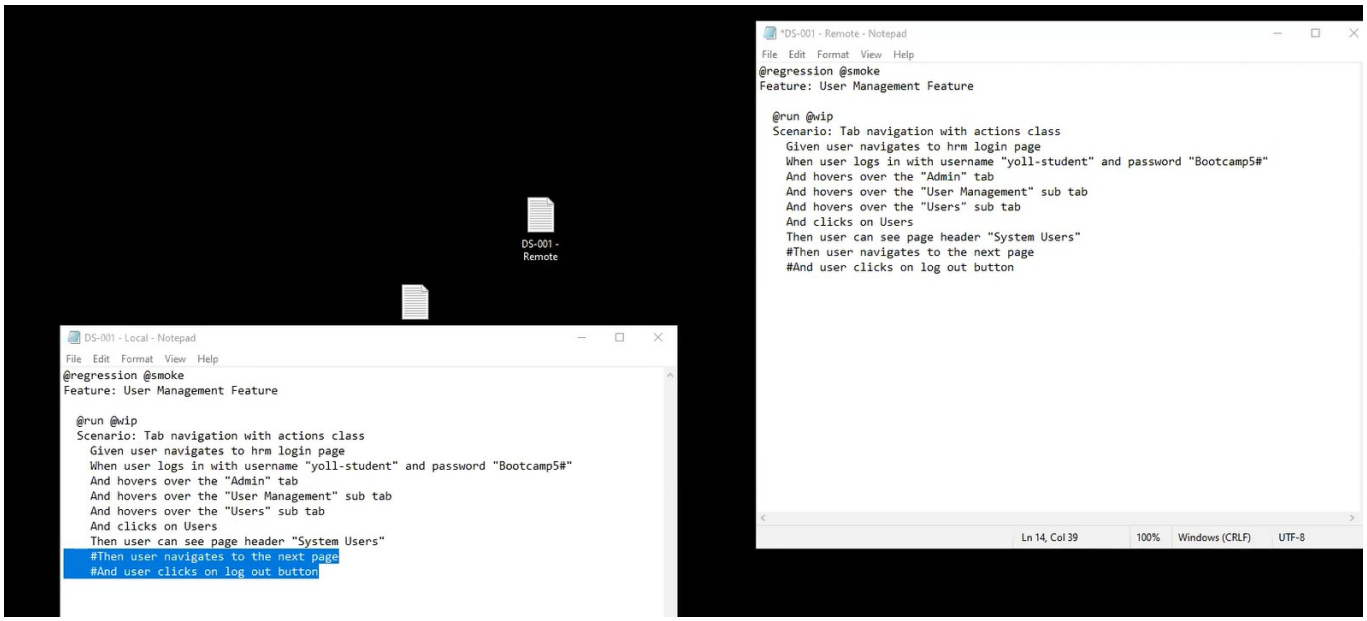
Then I Commit and push it.

Then I see an update From the Local DS-001 to the Remote DS-001 (You have to commit and Push)

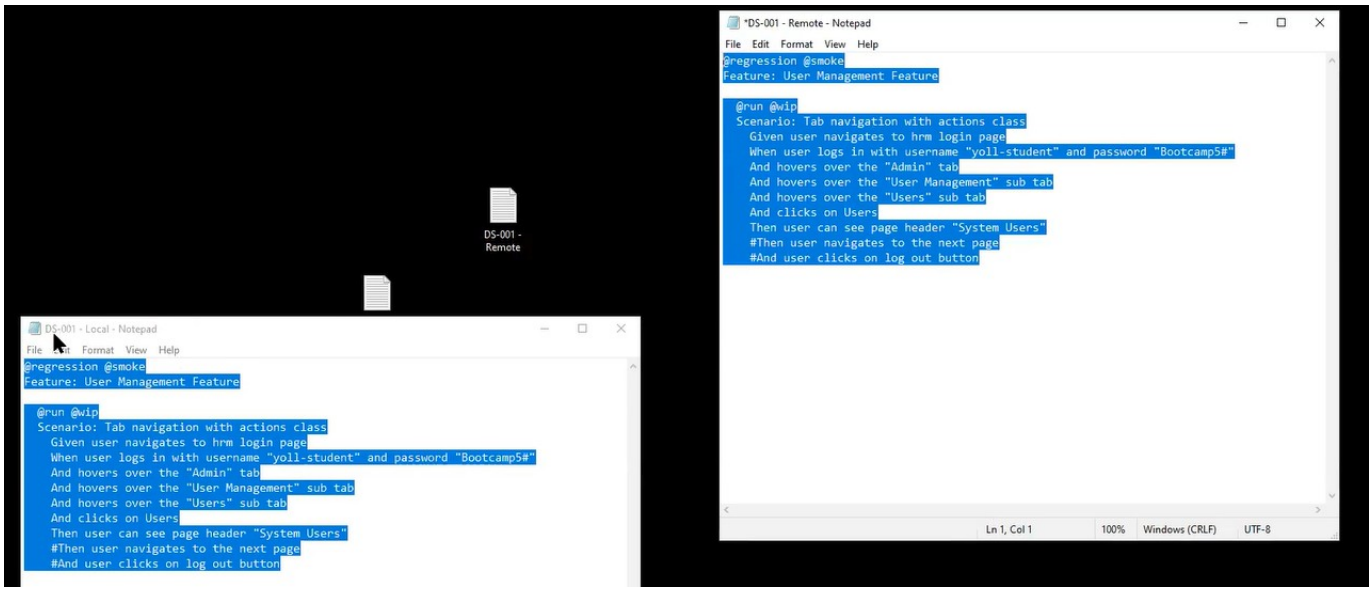
Then I do a change from the Local DS-001 to another Local DS-002 (You must merge and push, but you do not neet to commit)

Final result: We can a change from Local DS-002 (IntelliJ) to Remote DS-002 (GitHub)

1- Change on DS-001 Local:


DS-001 - Remote
DS-001 - Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
*DS-001 - Remote - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
Ln 14, Col 39
100%
Windows (CRLF)
UTF-8

2- After commit and Push from DS-001 Local to DS-001 Remote:

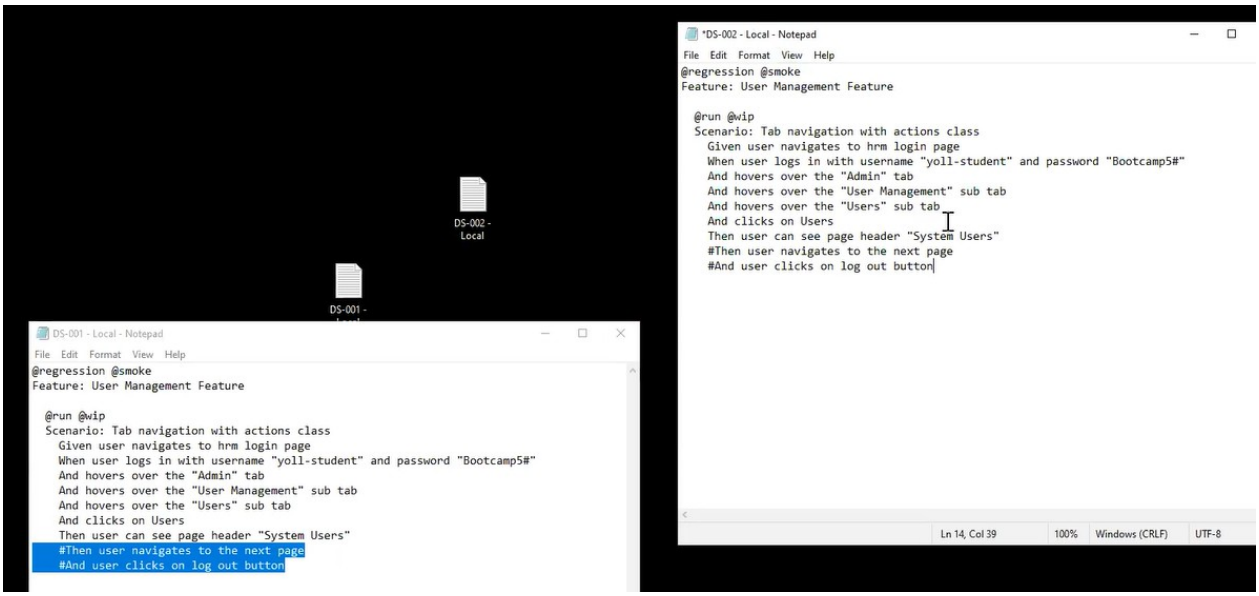

DS-001 - Remote
DS-001 - Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
*DS-001 - Remote - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8

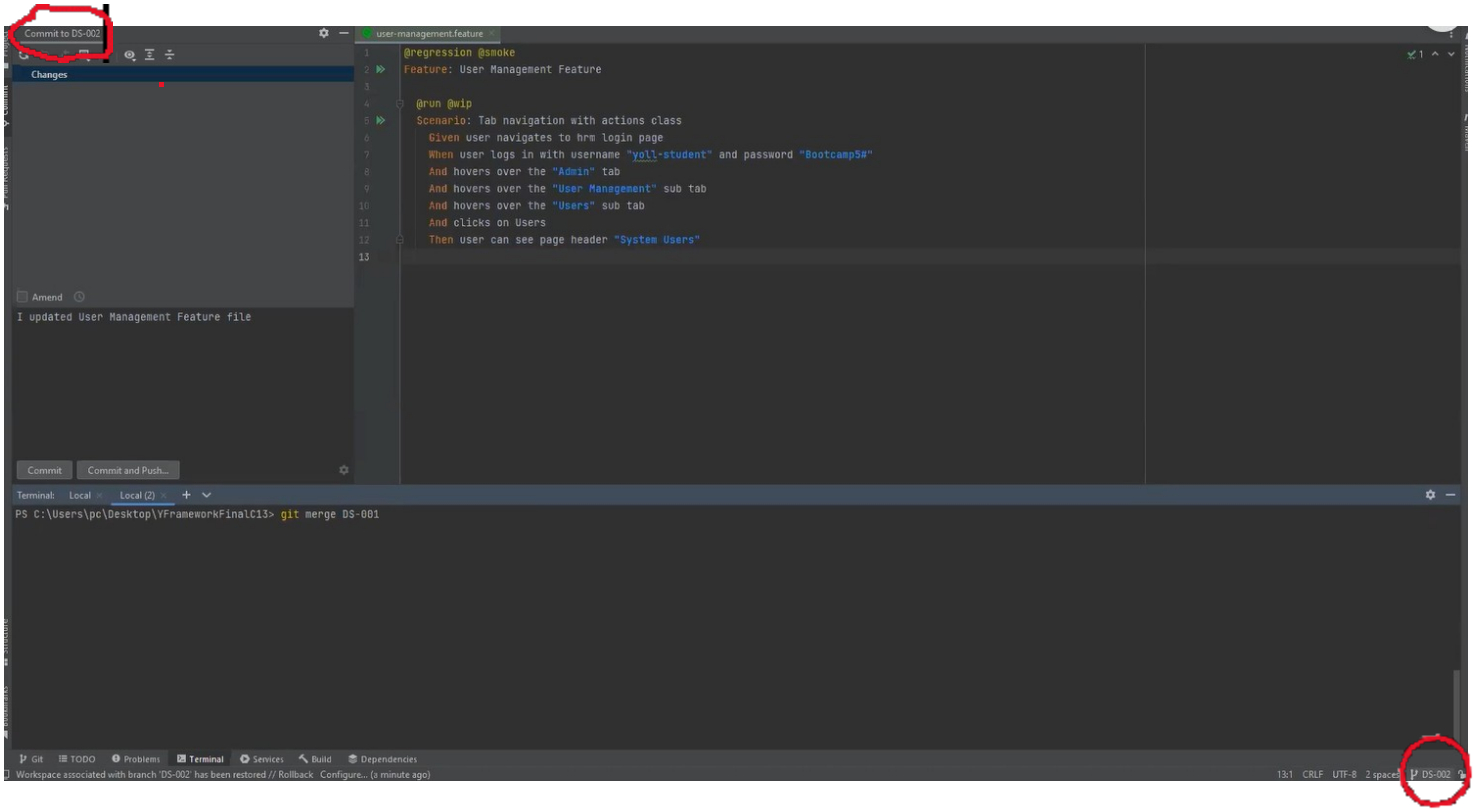
3- Change from the Local DS-001 to another local: Local DS-002

- **How to do it in Intellij ?**

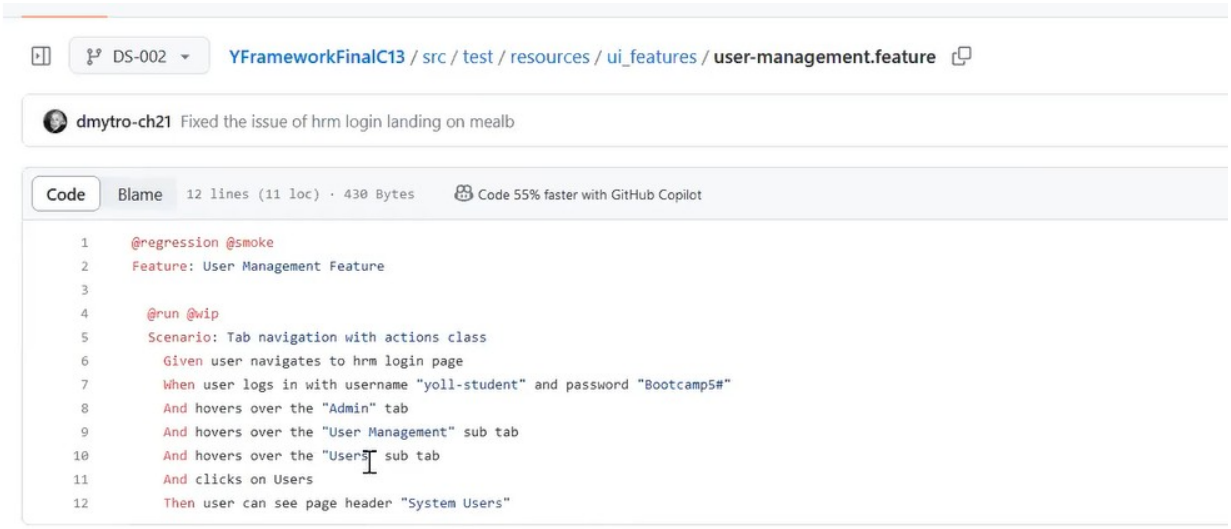
  Go to the Local DS-002 / execute the command:

  git merge DS-001


DS-002 - Local
DS-001 -
DS-001 - Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
*DS-002 - Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
Ln 14, Col 39
100%
Windows (CRLF)
UTF-8

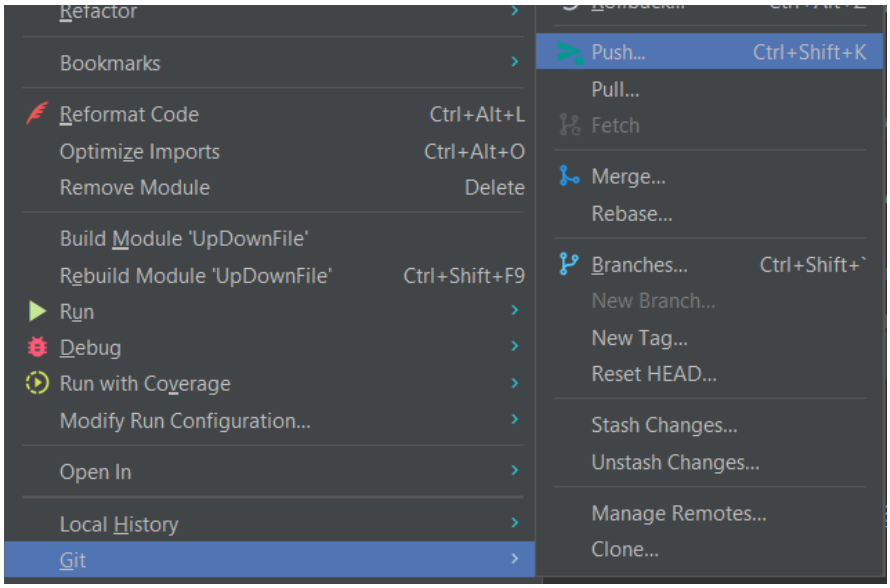

Commit to DS-002
Changes
Amend
I updated User Management Feature file
Commit
Commit and Push...
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
Terminal: Local Local (2)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git merge DS-001
Git TODO Problems Terminal Services Build Dependencies
Workspace associated with branch 'DS-002' has been restored // Rollback Configure... (a minute ago)
13:1 CRLF UTF-8 2 spaces DS-002

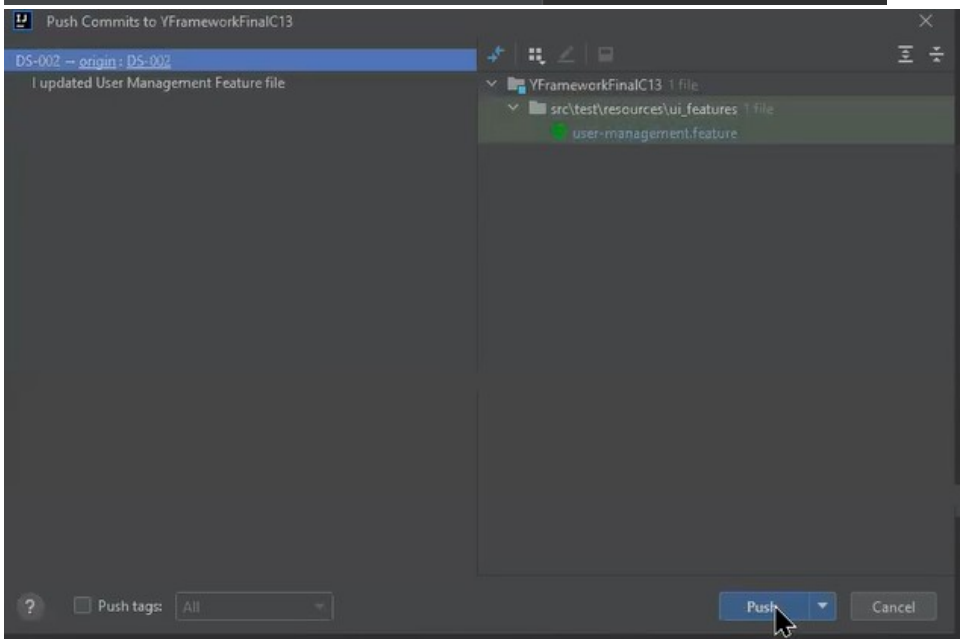
On GitHub, we cannot see the changes yet on GitHub !


DS-002
YFrameworkFinalC13 / src / test / resources / ui_features / user-management.feature
dmytro-ch21 Fixed the issue of hrm login landing on mealb
Code
Blame
12 lines (11 loc) · 430 Bytes
Code 55% faster with GitHub Copilot
1 @regression @smoke
2 Feature: User Management Feature
3
4 @run @wip
5 Scenario: Tab navigation with actions class
6 Given user navigates to hrm login page
7 When user logs in with username "yoll-student" and password "Bootcamp5#"
8 And hovers over the "Admin" tab
9 And hovers over the "User Management" sub tab
10 And hovers over the "Users" sub tab
11 And clicks on Users
12 Then user can see page header "System Users"

After, we Push, we can see the changes on lines 13 and 14:
We do not need to Commit at this case because it is local, and Local DS-001 has already been Commited, Just use : Push.
How to push ?
Execute the command: git push
Or by : Right click on the project / Git / Push

```
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git merge DS-001
Updating c3d2f44..637a115
Fast-forward
 src/test/resources/ui_features/user-management.feature | 1 +
 1 file changed, 1 insertion(+)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AliBoozkurt/YFrameworkFinalC13.git
   c3d2f44..637a115  DS-002 -> DS-002
PS C:\Users\pc\Desktop\YFrameworkFinalC13>
```





Refresh the page !

It has been merged now !

And you can Result : We can see Change from the Local DS-002 to another local: Remote DS-002 on GitHub

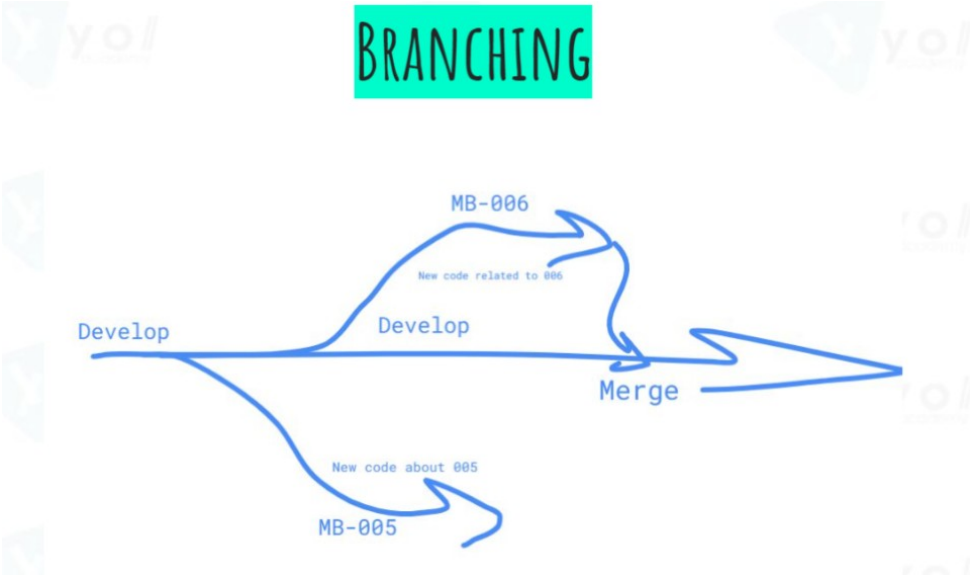
```
13          #Then user navigates to the next page
14          #And user clicks on log out button
```

DS-002 YFrameworkFinalC13 / src / test / resources / ui_features / **user-management.feature**

Ali B I updated User Management Feature file

Code Blame 14 lines (13 loc) · 511 Bytes Code 55% faster with GitHub Copilot

```
1   @regression @smoke
2   Feature: User Management Feature
3
4     @run @wip
5     Scenario: Tab navigation with actions class
6       Given user navigates to hrm login page
7       When user logs in with username "yoll-student" and password "Bootcamp5#"
8       And hovers over the "Admin" tab
9       And hovers over the "User Management" sub tab
10      And hovers over the "Users" sub tab
11      And clicks on Users
12      Then user can see page header "System Users"
13      #Then user navigates to the next page
14      #And user clicks on log out button
```



Explanation of the Diagram:

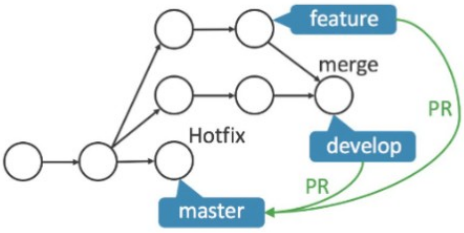
1. **Main Development Branch ("Develop")**
   - The `Develop` branch represents the primary development workflow.
   - New features or fixes are created as separate branches from `Develop`.
2. **Feature Branches (`MB-005` and `MB-006`)**
   - `MB-005`: A separate branch created for a feature or fix related to issue/task **005**.
   - `MB-006`: Another branch created for issue/task **006**, developed separately from `MB-005`.
   - Both branches contain new code specific to their respective tasks.
3. **Merging Back to `Develop`**

- Once the work in `MB-006` is complete, it is merged back into `Develop`.
- `MB-005` does not show a merge in this diagram, meaning it may still be in progress or handled separately.

Key Takeaways from the Diagram:

- **Branching isolates work** on different tasks/features.
- **Developers work on separate branches** (`MB-005` and `MB-006`) before merging changes.
- **Merge integrates changes** from feature branches back into `Develop`.

- Pull Request *(Another name Merge Request)*
- When you are about to merge a branch to master or another important branch, you want it to be **reviewed** by team lead or fellow team member.
- Git provides a mechanism for this moderation, which is called Pull Requests.

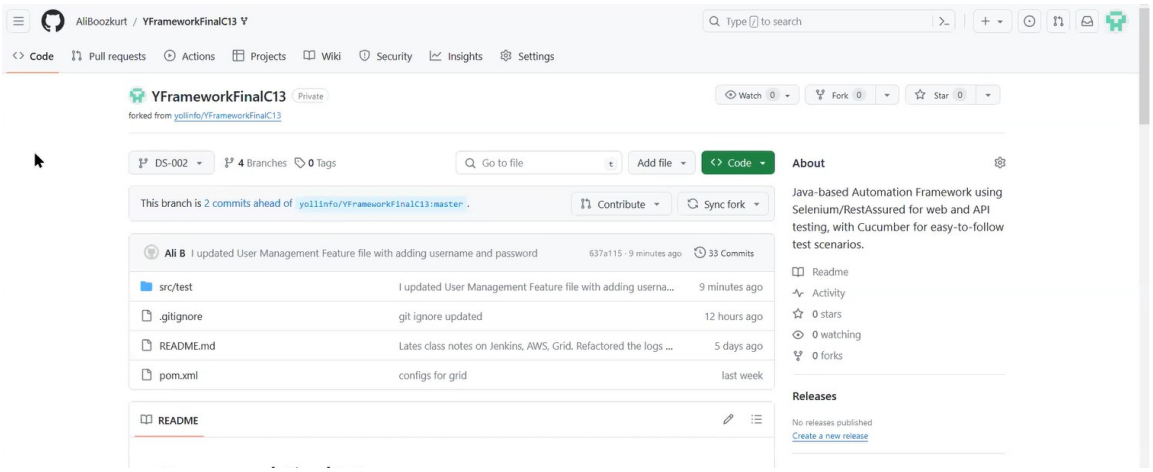


- **Pull requests** let you tell others about changes you are about to merge to a GitHub repository.
- Pull Requests are created via **Github Web site**.
- Once a pull request is created, team lead can **review the set of changes**, comment on the changes.
- After reviewed, they can either approve and merge the PR, or not merge it if they see an issue.
- After PR is merged, then PR gets closed.

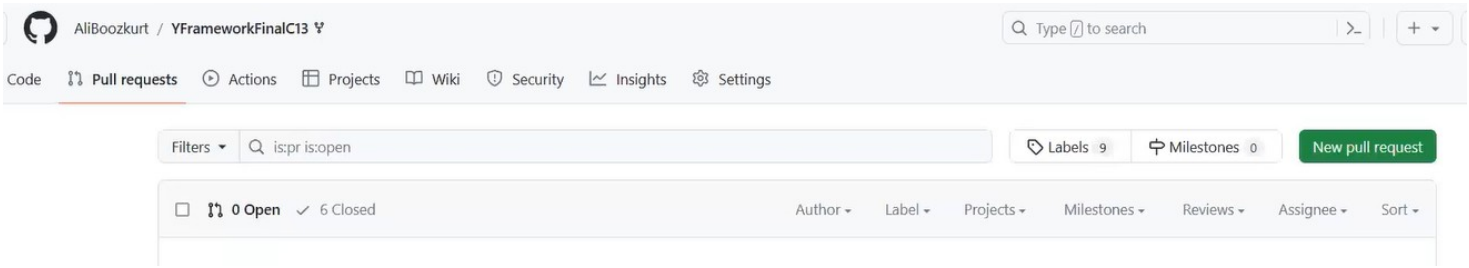
Task

Using PR from Local DS-001 to dev-Remote.
I need to merge Local DS-001 to dev-Remote.
I have to create a request on GitHub first to be reviewed by the team/leader/member. Othewise, I will be rejected !

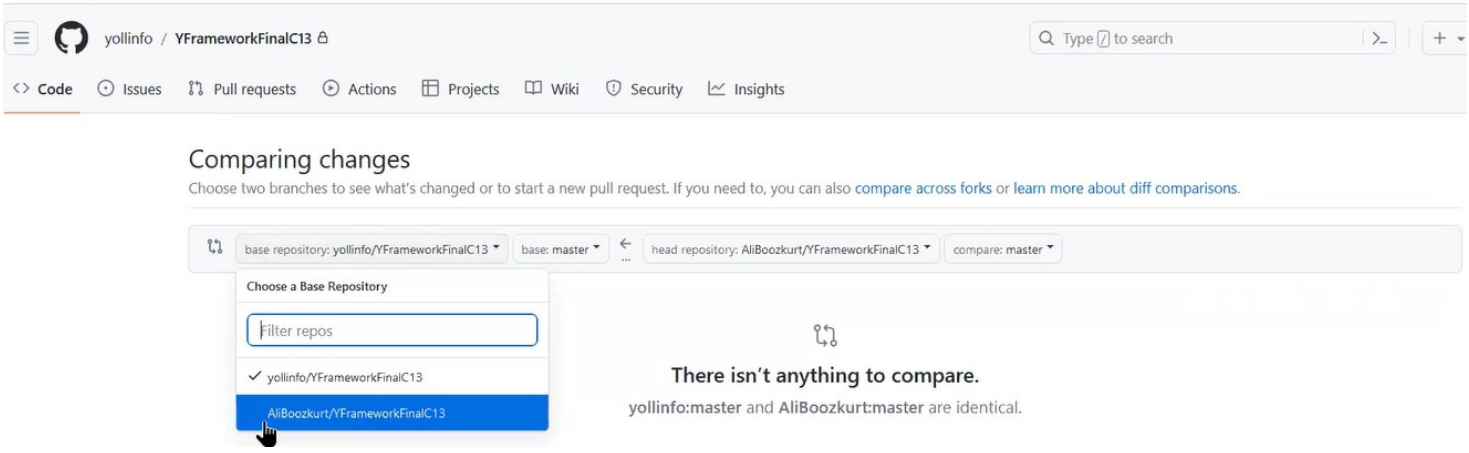
- **Checkout to the Local DS-001 in IntelliJ.**
- **Click on Pull requests button on GitHub. Make sure you have your name on your account.**

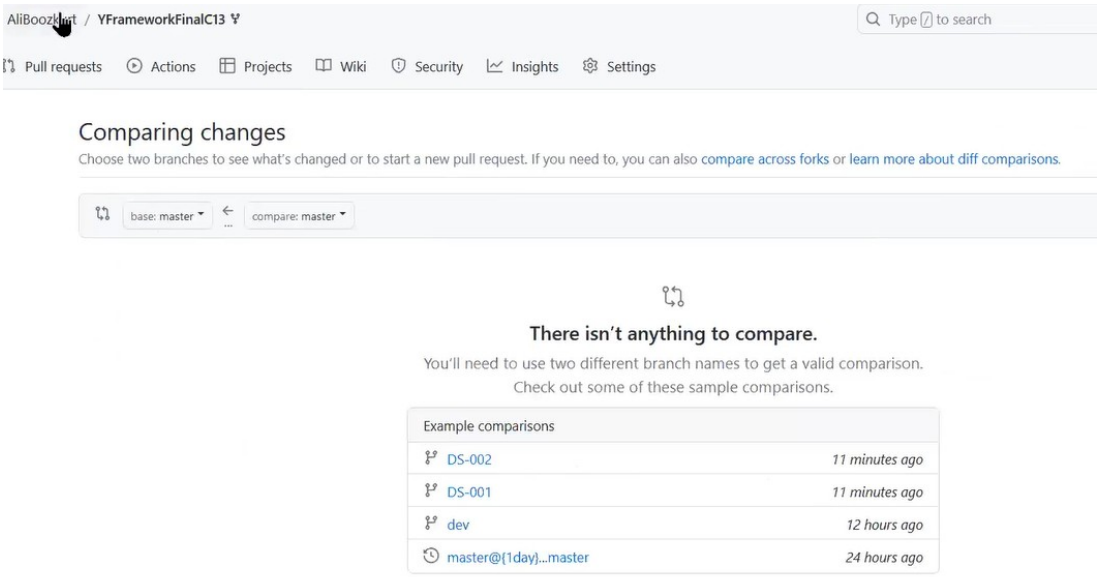


- **Click on New pull request button.**

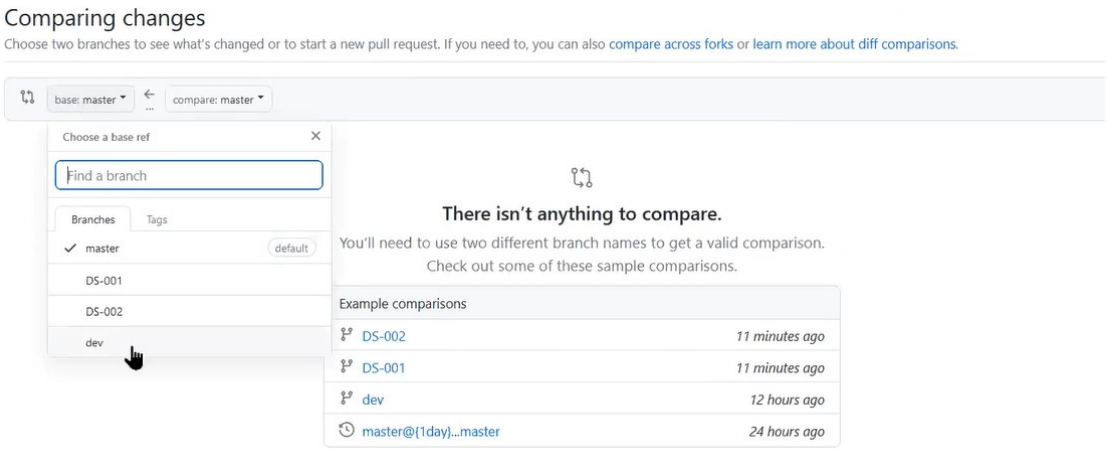


- **Choose a Base Repository containing your name.**

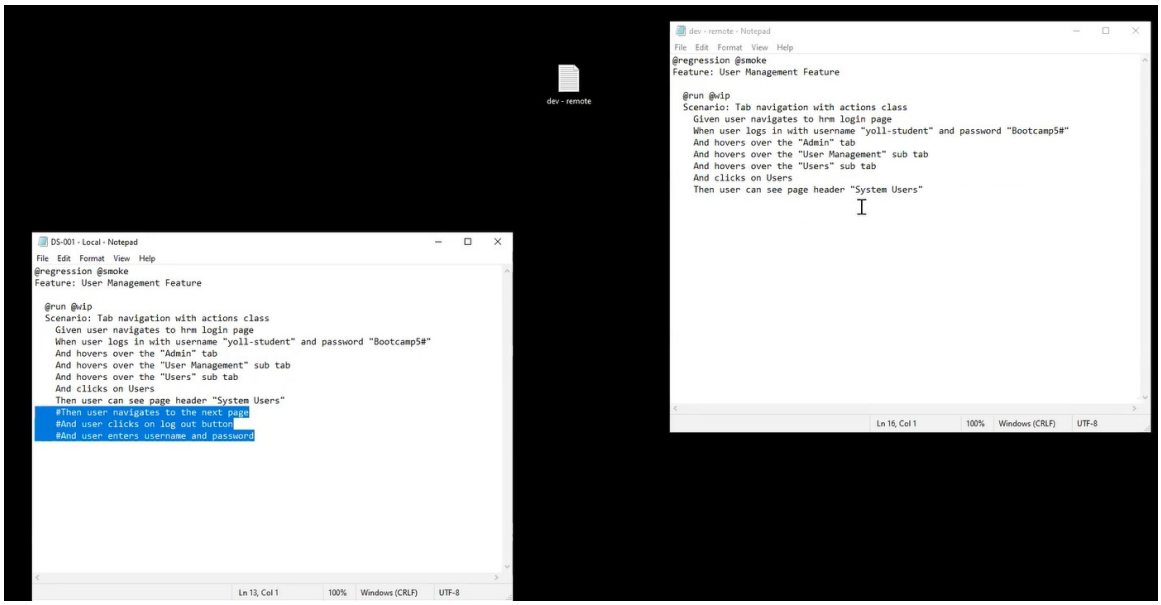


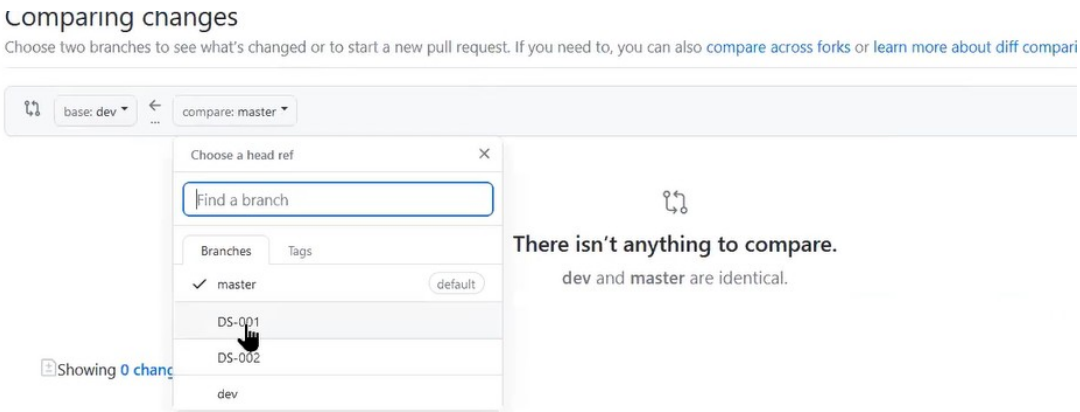


- **Choose your destination. In our case : dev.**

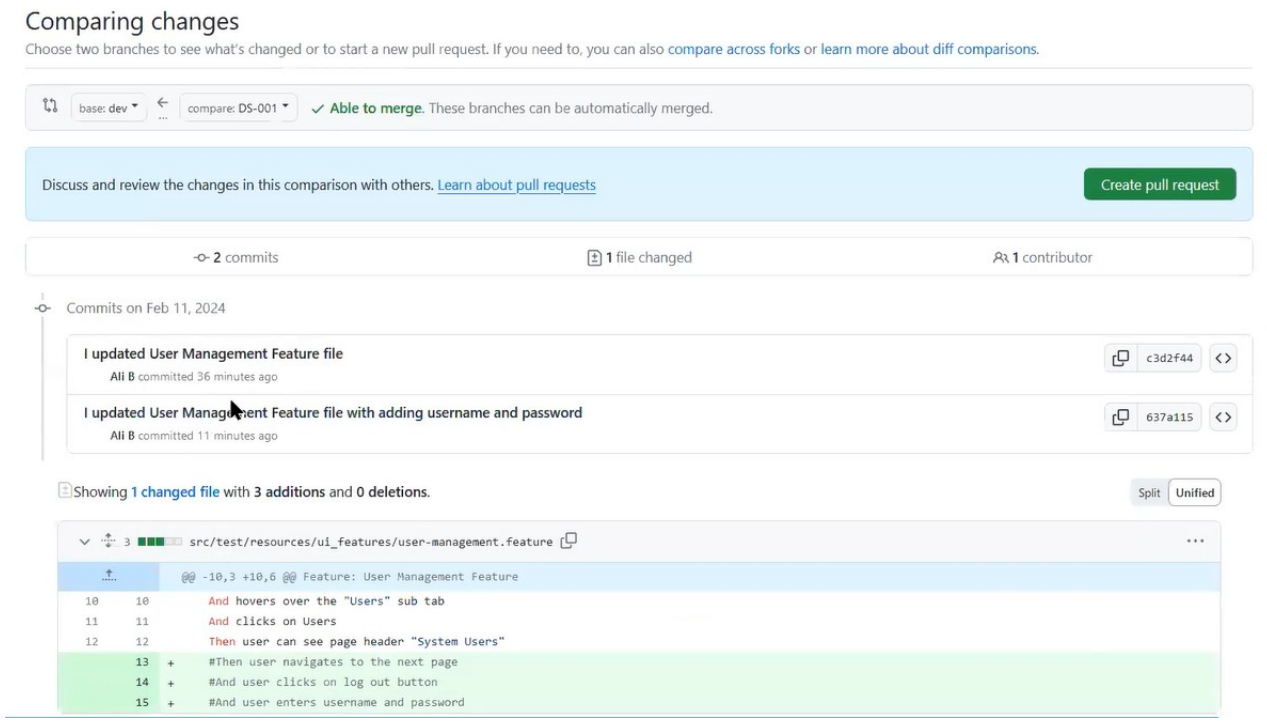


- **I want to merge Local DS-001 to dev:**

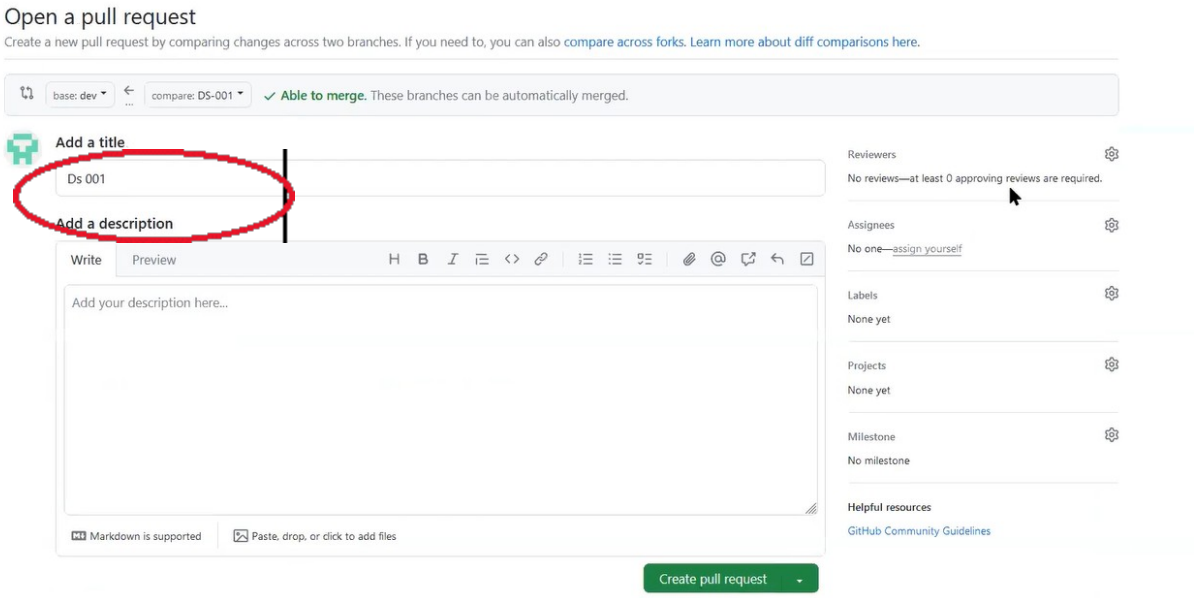

dev - remote
dev - remote - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
Ln 16, Col 1
100%
Windows (CRLF)
UTF-8
DS-001 - Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
Ln 13, Col 1
100%
Windows (CRLF)
UTF-8

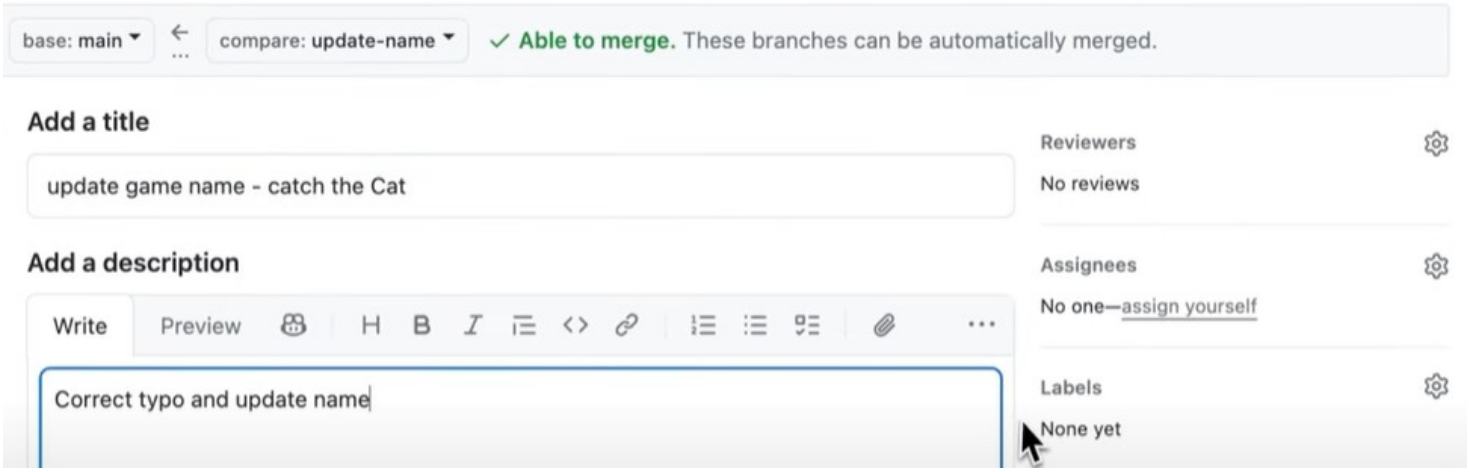


- **The green color is what will be merged**

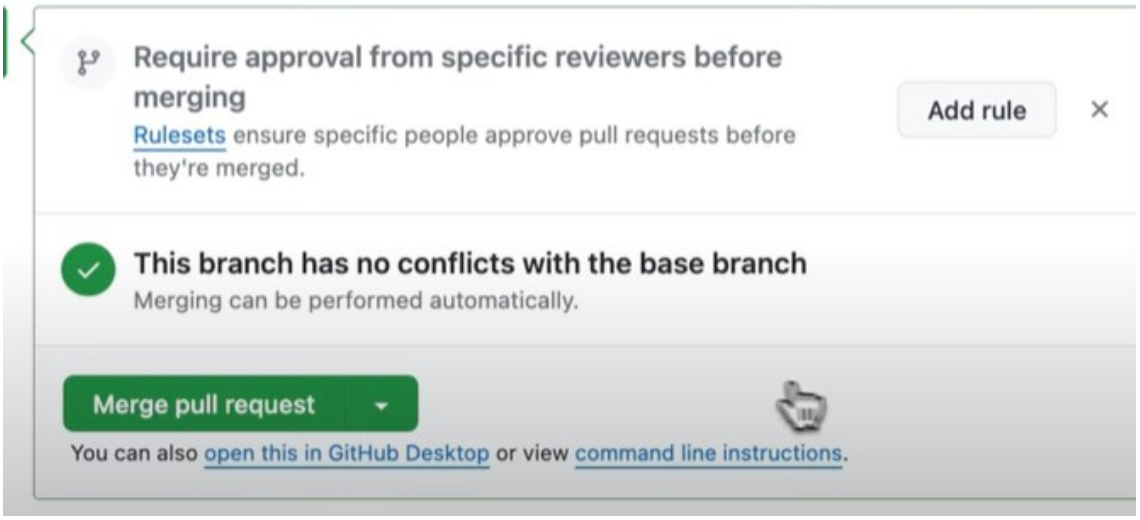

Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base: dev
compare: DS-001
Able to merge. These branches can be automatically merged.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request
2 commits
1 file changed
1 contributor
Commits on Feb 11, 2024
I updated User Management Feature file
Ali B committed 36 minutes ago
c3d2f44
I updated User Management Feature file with adding username and password
Ali B committed 11 minutes ago
637a115
Showing 1 changed file with 3 additions and 0 deletions.
Split
Unified
src/test/resources/ui_features/user-management.feature
@@ -10,3 +10,6 @@ Feature: User Management Feature
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password

- **Click on the Green Create pull request button after putting a title and a description:**





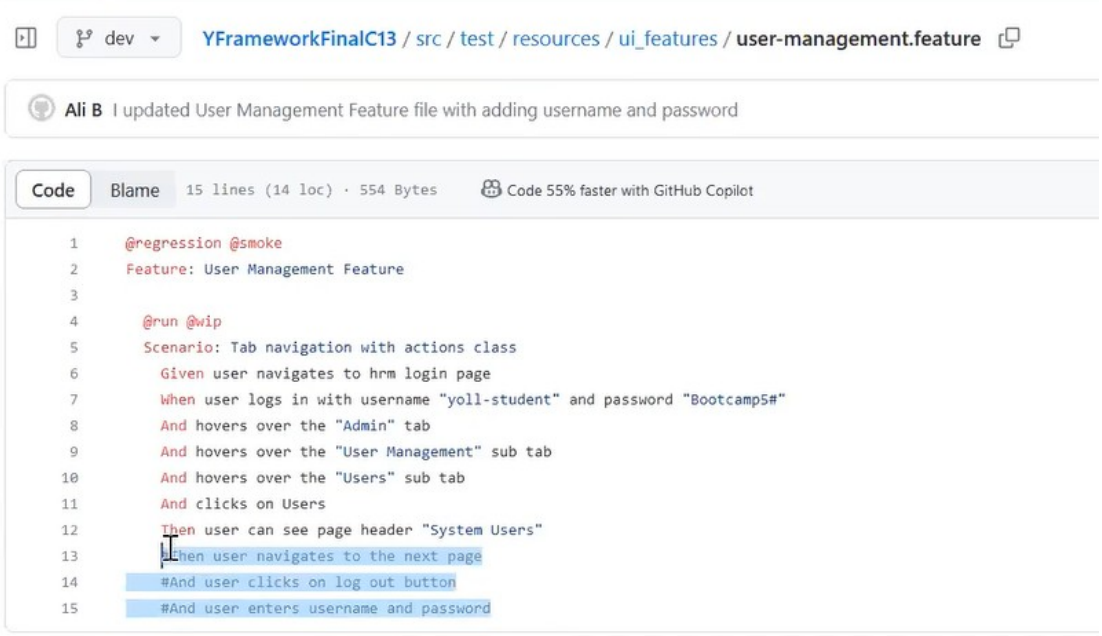
Wait for the approval, then you can merge pull requests.


Require approval from specific reviewers before merging
Rulesets ensure specific people approve pull requests before they're merged.
Add rule
This branch has no conflicts with the base branch
Merging can be performed automatically.
Merge pull request
You can also open this in GitHub Desktop or view command line instructions.

Click on Confirm Merge

Add more commits by pushing to the ds-001 branch on AliBoozkurt/YFrameworkFinalC13.
Merge pull request #7 from AliBoozkurt/DS-001
Ds 001
This commit will be authored by 43423583+AliBoozkurt@users.noreply.github.com
Confirm merge
Cancel

If you want to check it out the changes on dev:
Go to the feature file


dev
YFrameworkFinalC13 / src / test / resources / ui_features / user-management.feature
Ali B I updated User Management Feature file with adding username and password
Code
Blame
15 lines (14 loc) · 554 Bytes
Code 55% faster with GitHub Copilot
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password

If you are the only person who can review the changes, you do not need to create a pull request.

There is another way to create a pull request by using Bitbucket

MERGE CONFLICTS

- Git is smart to auto-merge your changes but there are times GIT can't automerge because **same lines have been touched by different developers.**
- Git simply **cannot** know what is correct. Git will then mark the file as having a **conflict** - which you'll have to solve before you can continue your work.
- Human intervention is needed. This process is called **Resolving Merge Conflicts.**

Can occur:

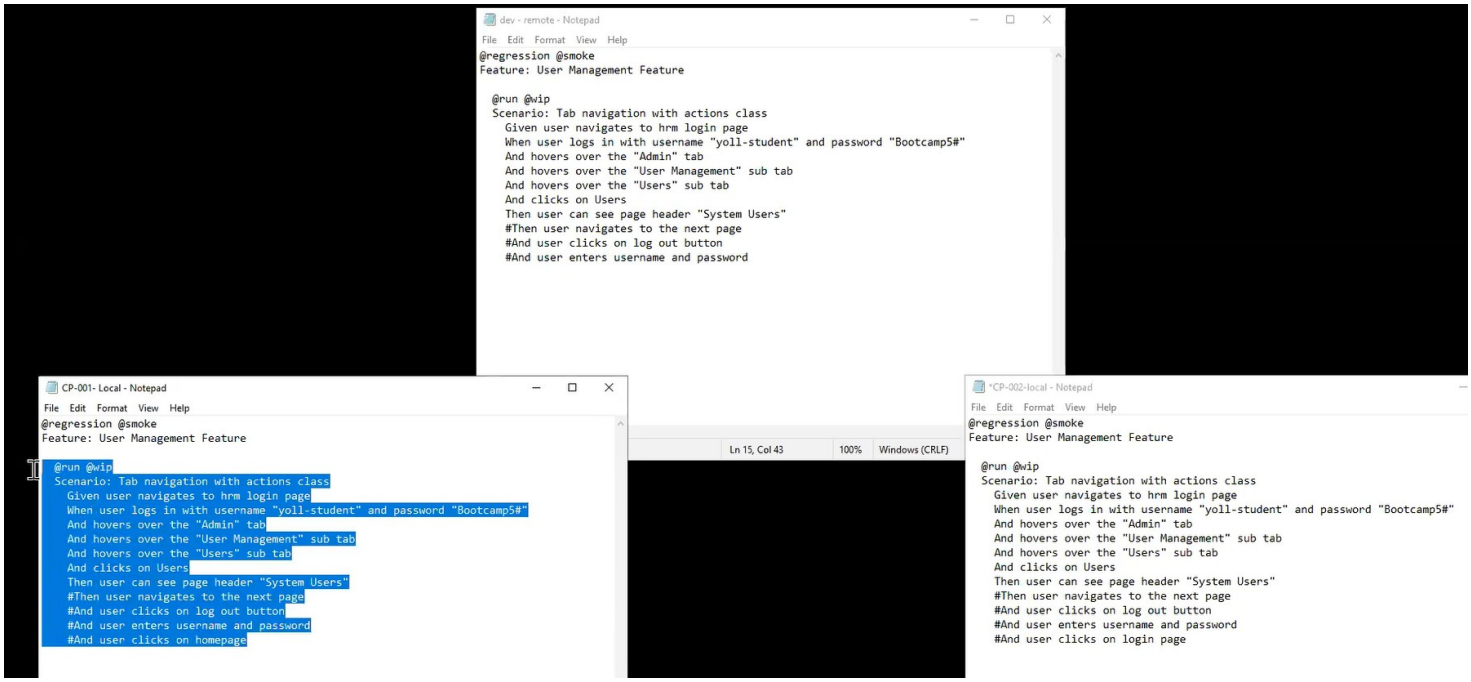
- When you **git pull** changes. → remote version of same branch may have same line of code changed by others.
- When you **merge branches** → remote target branch may have same line of code changed by other.

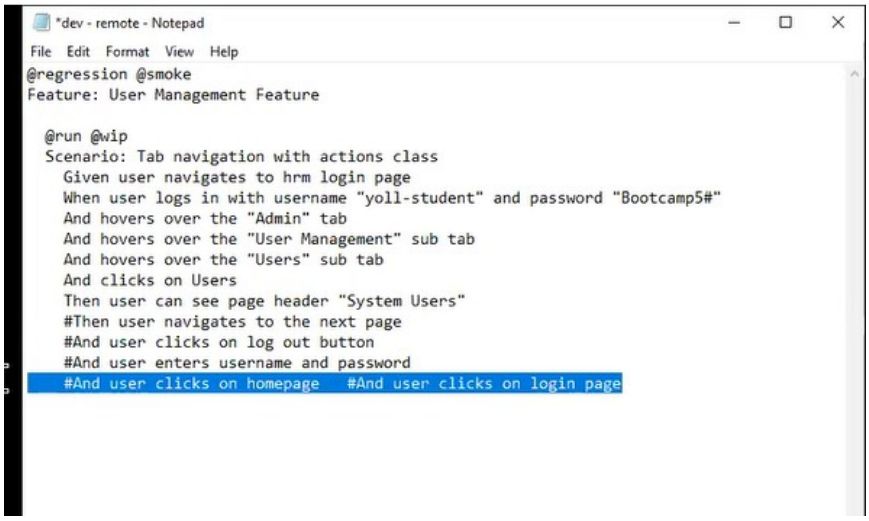
Example of conflict merge:

CP-001 To dev-remote by adding the last line: #And user clicks on homepage
CP-002 To dev-remote by adding the last line: #And user clicks on Login page

Result on dev is confusing, we need to decide which one:

#And user clicks on homepage #And user clicks on Login page

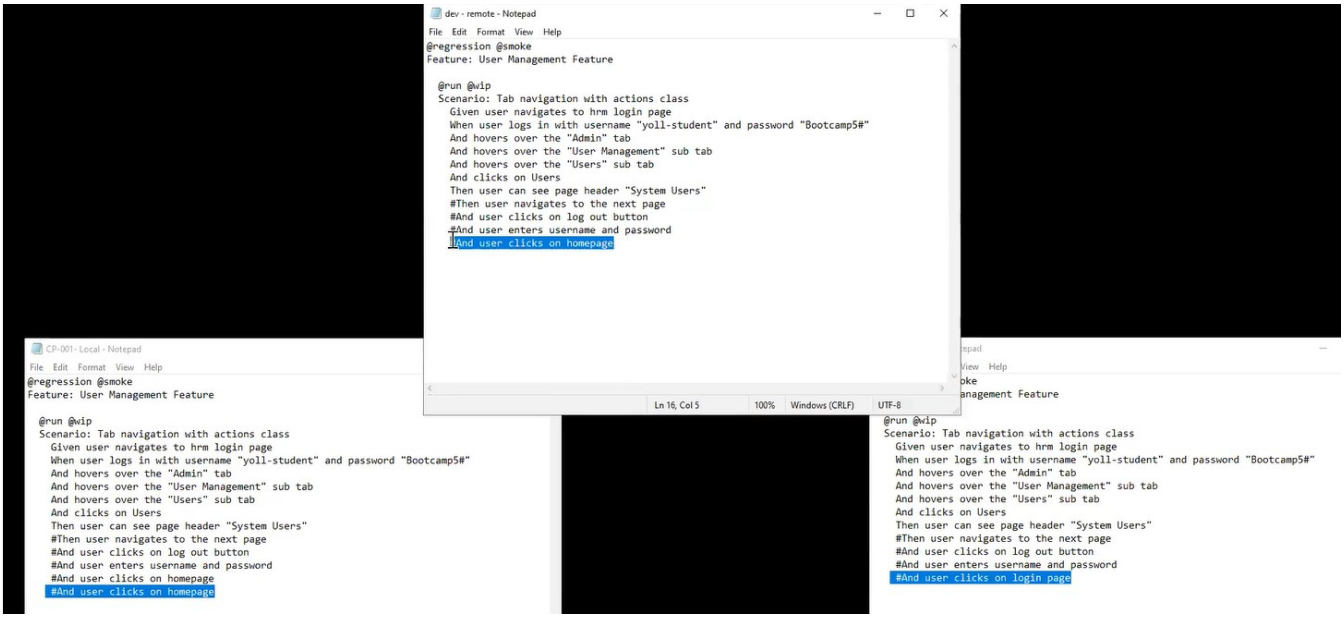

dev - remote - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
CP-001- Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage
Ln 15, Col 43
100%
Windows (CRLF)
*CP-002-local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on login page


*dev - remote - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage #And user clicks on login page

***Confusion:**

When CP-001 is merged with dev-remote by adding the last line, CP-001 is comapared with dev-remote line by line until the the last line is added.

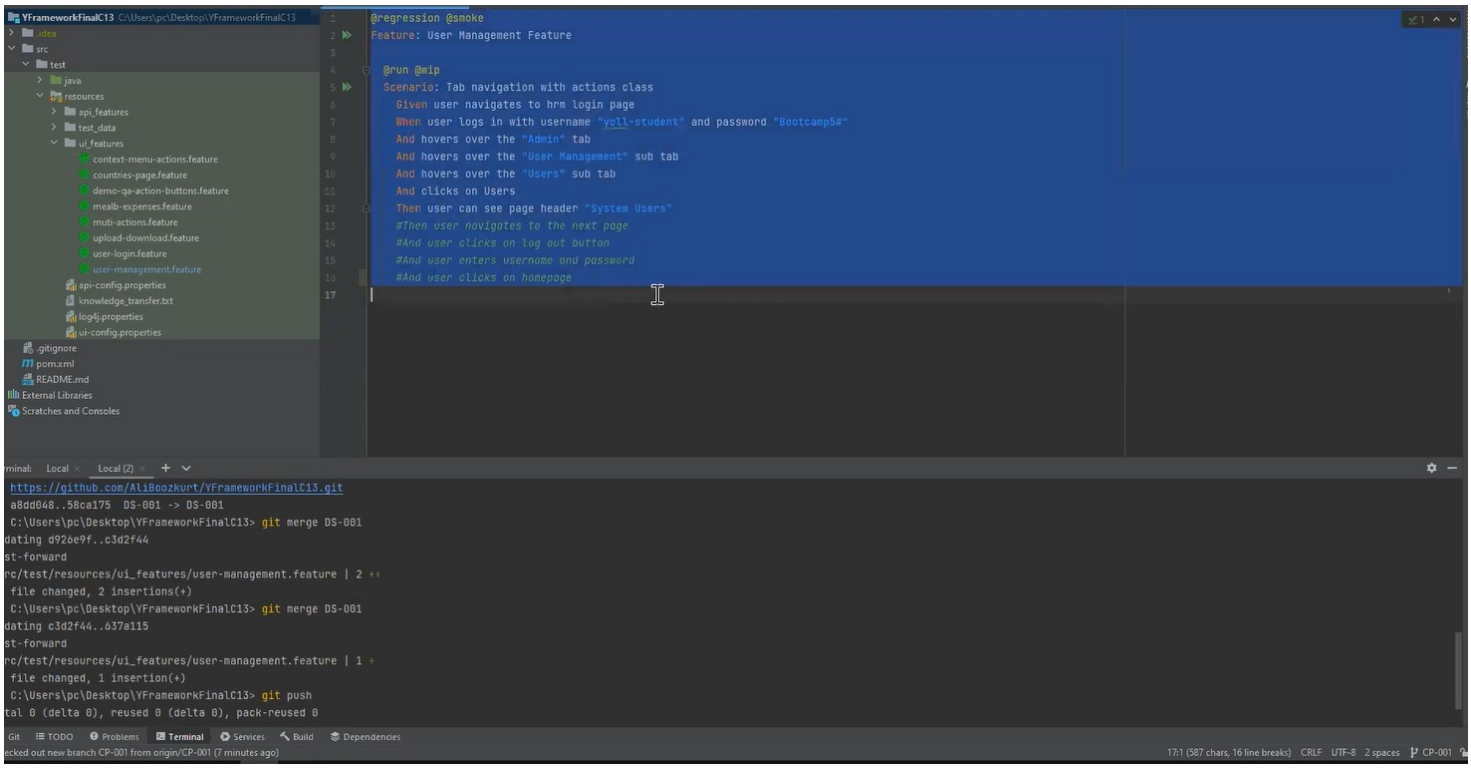
When CP-002 is merged with dev-remote by adding the last line, CP-002 is comapared with dev-remote line by line until the the last line of CP-002 and dev-remote are different. Which means there is a conflict merge and CP-002 is unable to be merged


dev - remote - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage
Ln 16, Col 5
100%
Windows (CRLF)
UTF-8
CP-001 - Local - Notepad
File Edit Format View Help
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage
#And user clicks on homepage
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on login page

In intelliJ:

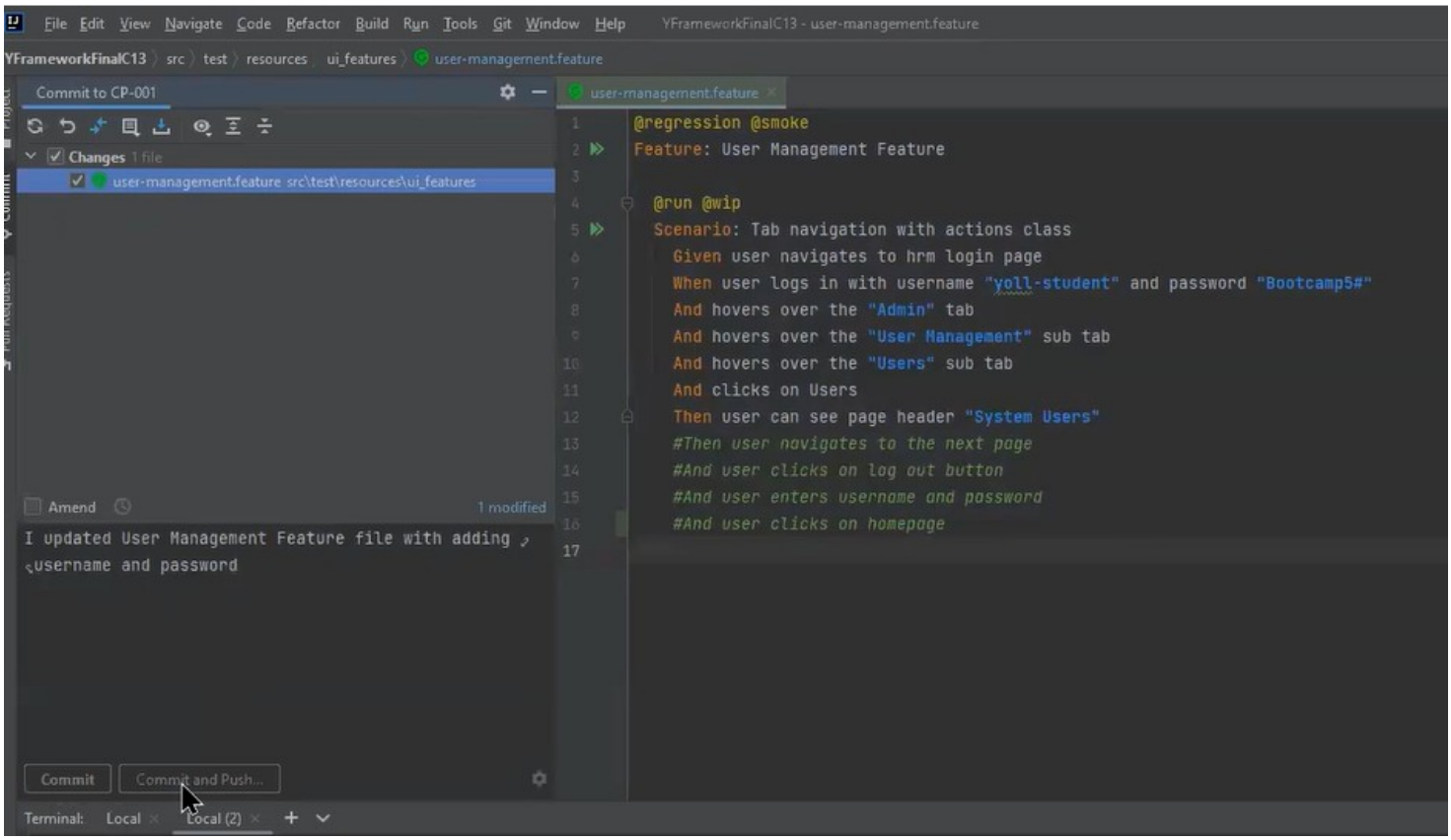
1- CP-001

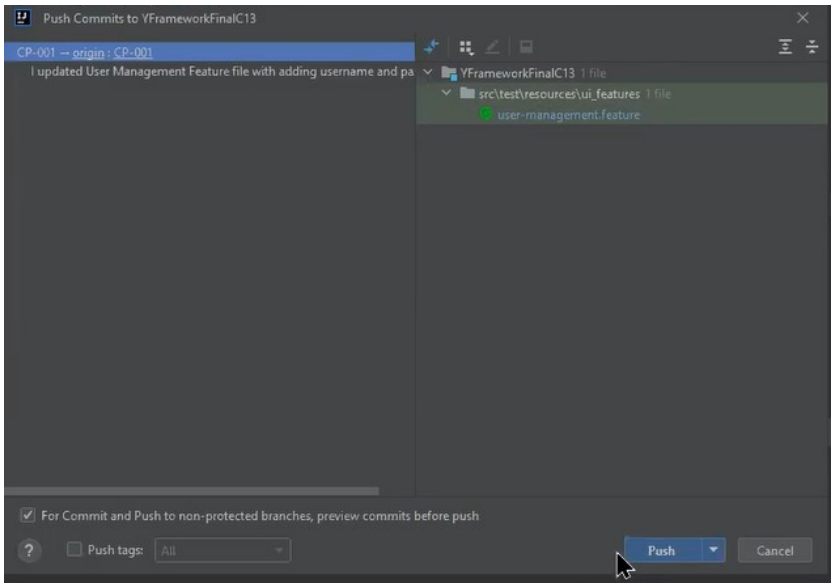
The last line contains: #And user clicks on homepage


@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage
https://github.com/AliBoozkurt/YFrameworkFinalC13.git
C:\Users\pc\Desktop\YFrameworkFinalC13> git merge DS-001
C:\Users\pc\Desktop\YFrameworkFinalC13> git merge DS-001
C:\Users\pc\Desktop\YFrameworkFinalC13> git push

Go to commit, Select Changes

Click on Commit and Push


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
YFrameworkFinalC13 - user-management.feature
Commit to CP-001
Changes 1 file
user-management.feature src\test\resources\ui_features
Amend
1 modified
I updated User Management Feature file with adding username and password
Commit
Commit and Push...
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage
Terminal: Local Local (2)

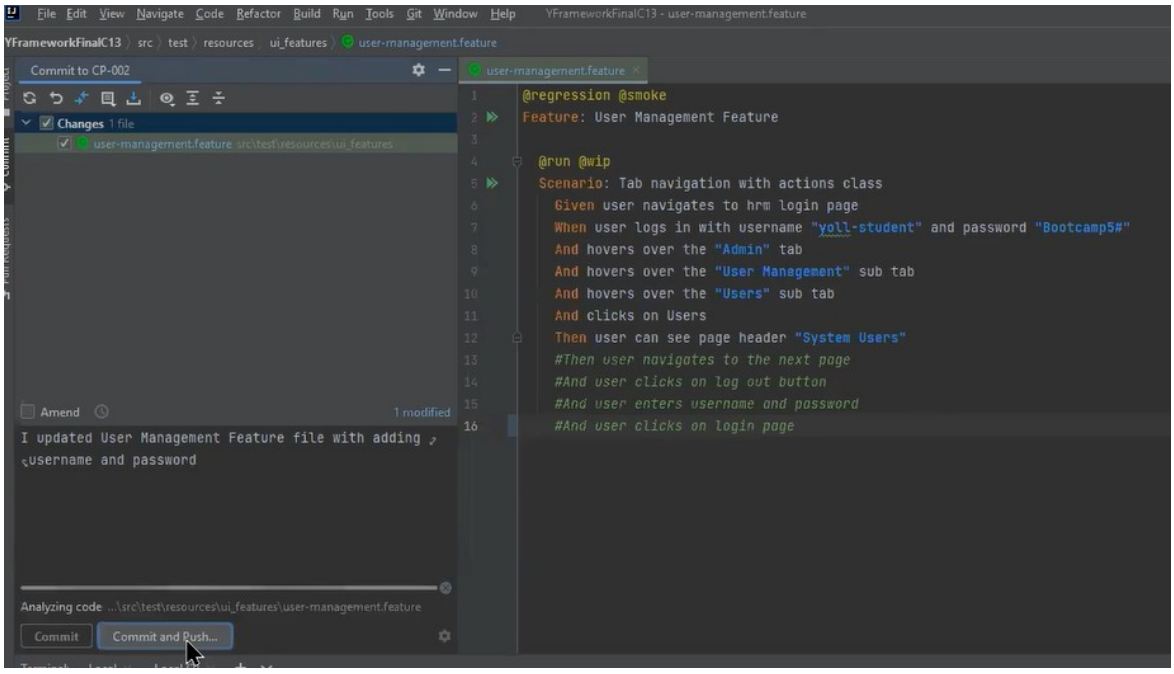


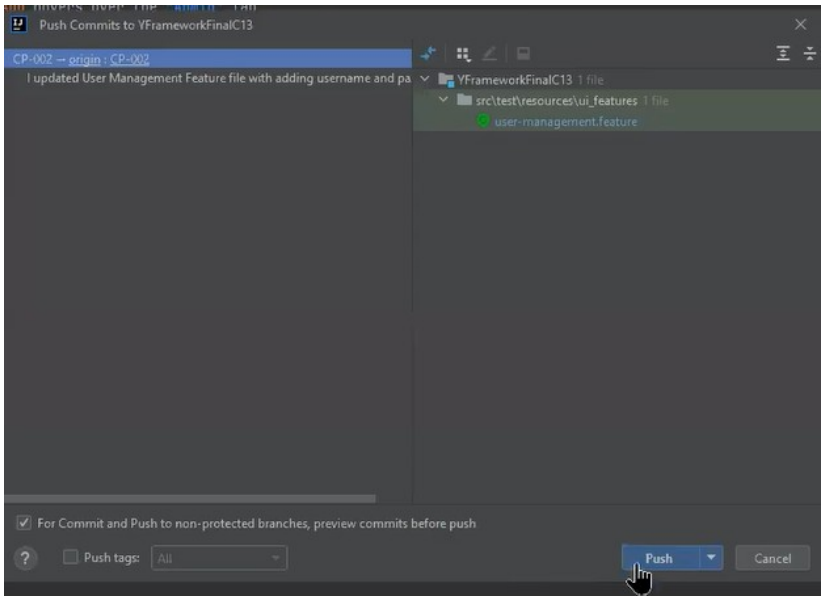
2- CP-001

The last line contains: #And user clicks on Login Page

Go to commit, Select Changes

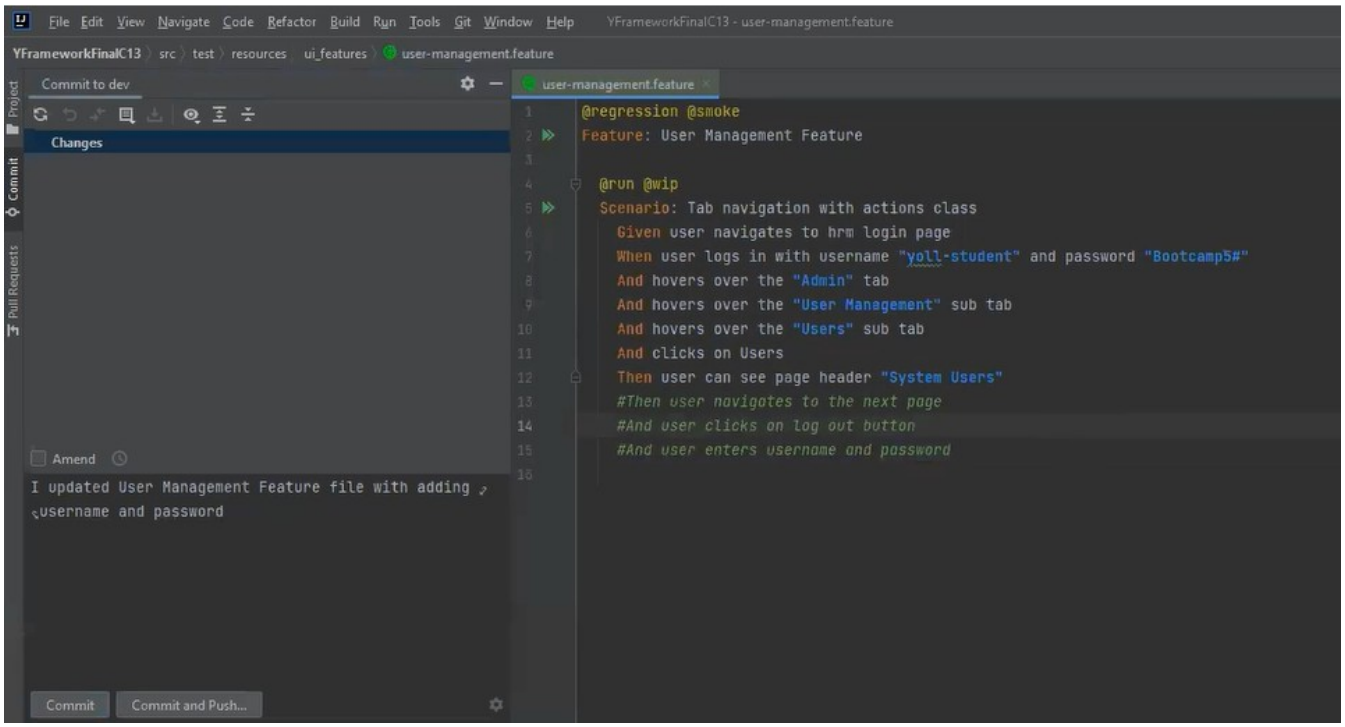
Click on Commit and Push


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
YFrameworkFinalC13 - user-management.feature
YFrameworkFinalC13 src test resources ui_features user-management.feature
Commit to CP-002
Changes 1 file
user-management.feature src\test\resources\ui_features
Amend
1 modified
I updated User Management Feature file with adding username and password
Analyzing code ...\src\test\resources\ui_features\user-management.feature
Commit
Commit and Push...
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on login page



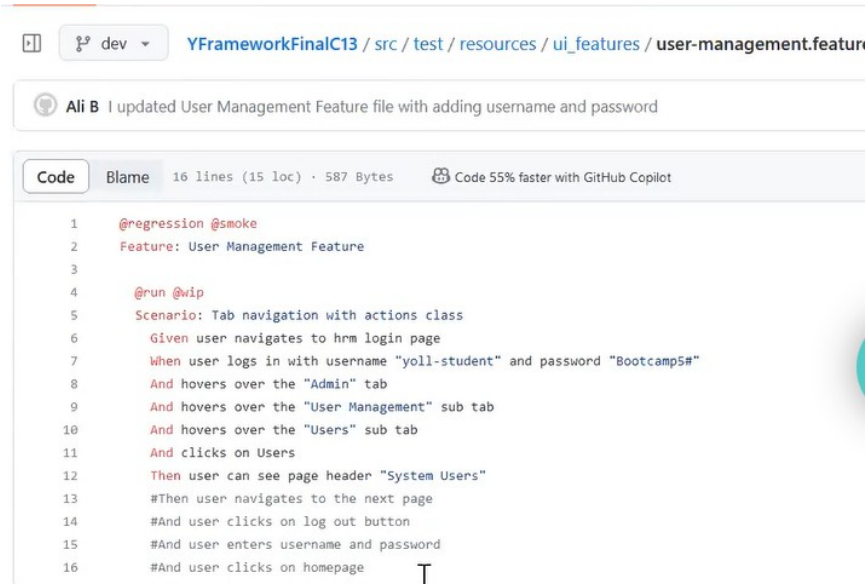
3- Dev:

Result: **No line from CP-001 or CP-002 is added to dev**


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
YFrameworkFinalC13 - user-management.feature
YFrameworkFinalC13 src test resources ui_features user-management.feature
Commit to dev
Changes
Amend
I updated User Management Feature file with adding username and password
Commit
Commit and Push...
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password

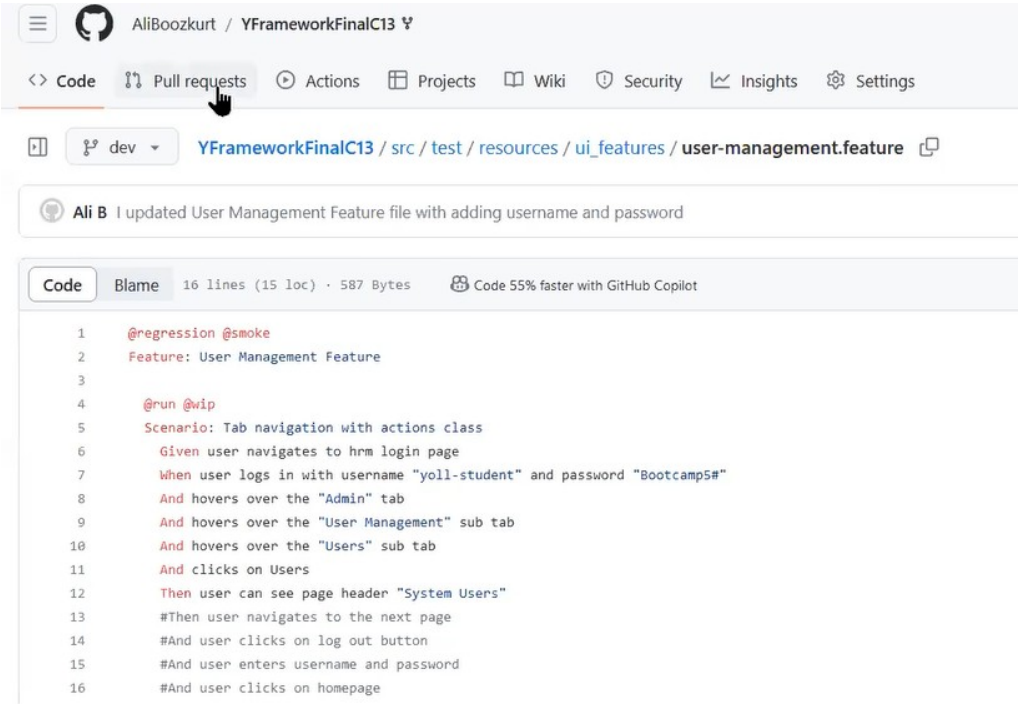
Let`s see what happened on GitHub on dev :

1- After Updating CP-001: the line #And user clicks on homepage is added

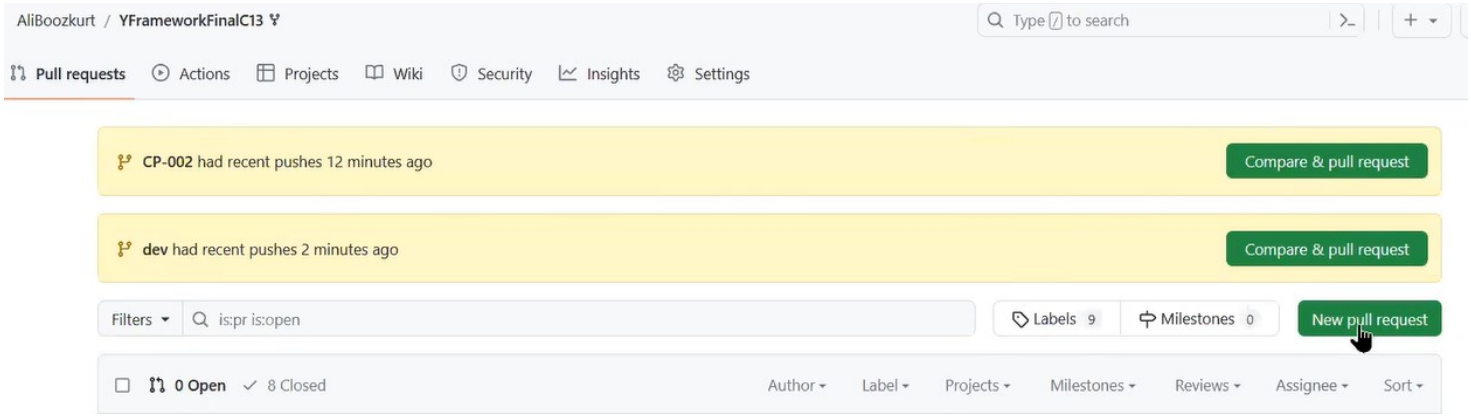

dev
YFrameworkFinalC13 / src / test / resources / ui_features / user-management.featur
Ali B I updated User Management Feature file with adding username and password
Code
Blame
16 lines (15 loc) · 587 Bytes
Code 55% faster with GitHub Copilot
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage

2- After Updating CP-002: the line #And user clicks on Login page is not added

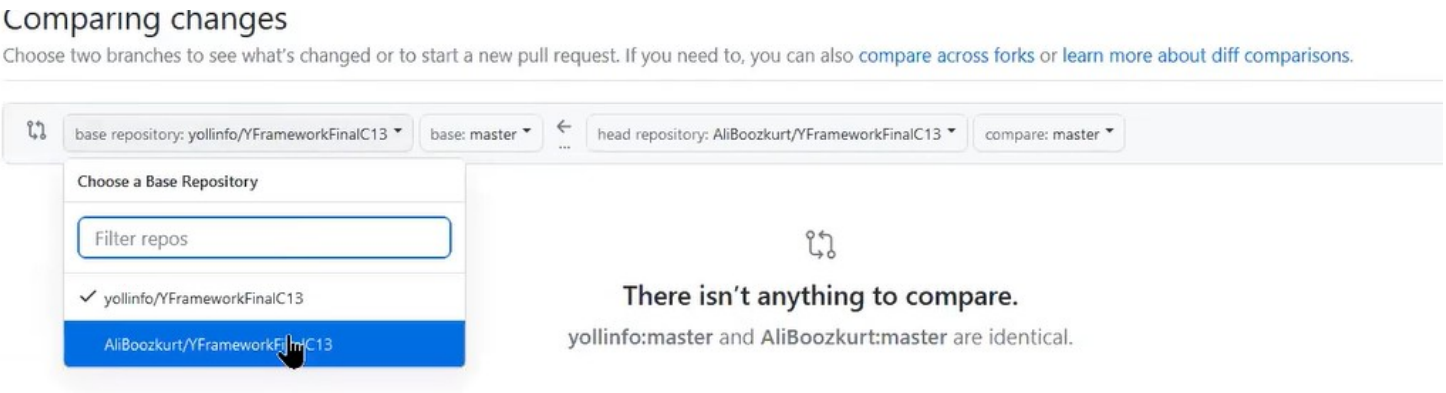
- **Click on: Pull Requests**


AliBoozkurt / YFrameworkFinalC13
Code
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
dev
YFrameworkFinalC13 / src / test / resources / ui_features / user-management.feature
Ali B I updated User Management Feature file with adding username and password
Code
Blame
16 lines (15 loc) · 587 Bytes
Code 55% faster with GitHub Copilot
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage

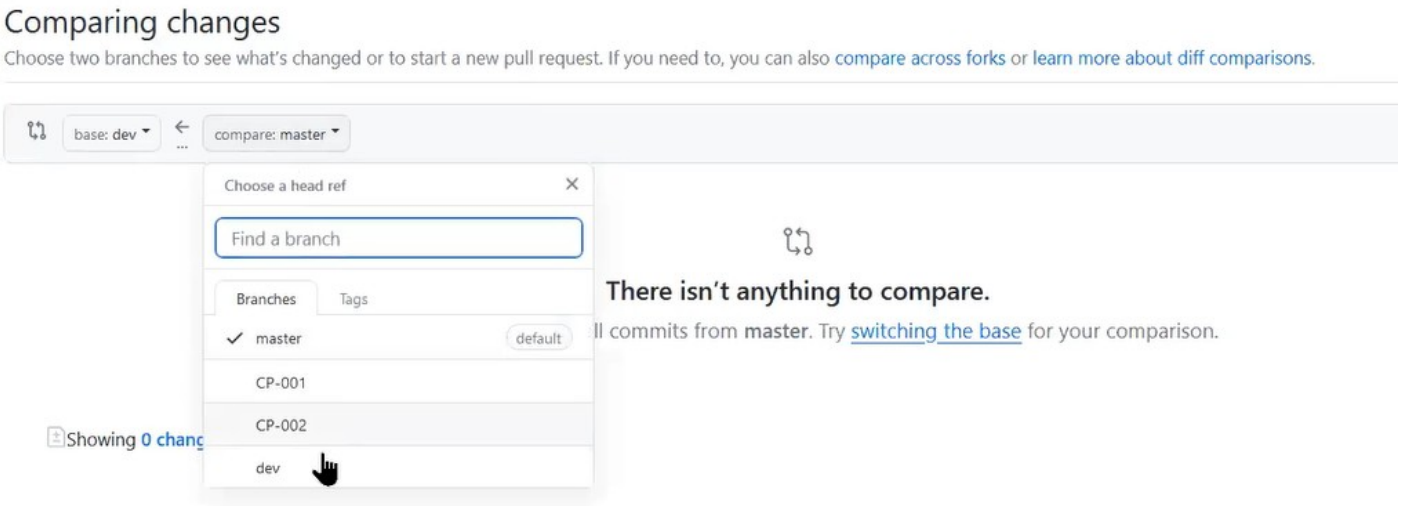
- **Click on: New Pull Request**



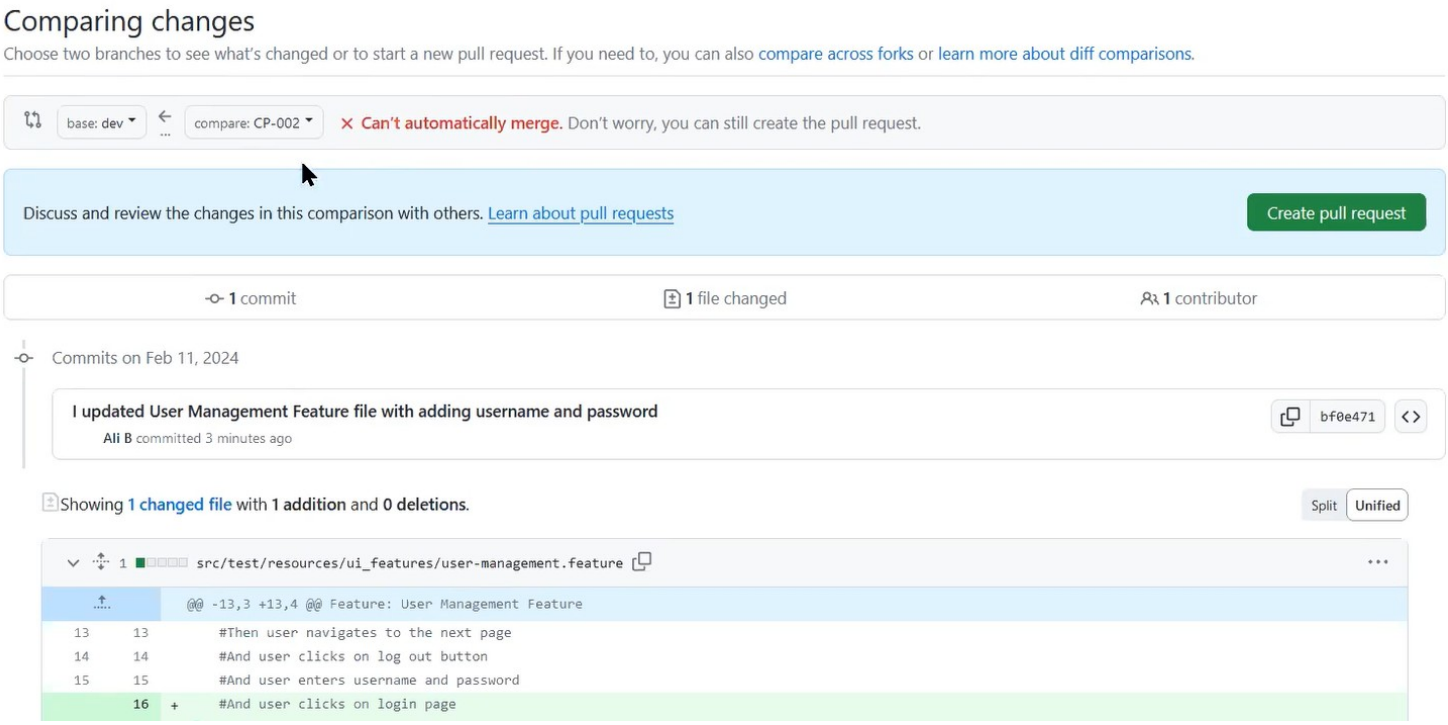
- **Choose your repository:**


Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base repository: yollinfo/YFrameworkFinalC13
base: master
head repository: AliBoozkurt/YFrameworkFinalC13
compare: master
Choose a Base Repository
Filter repos
yollinfo/YFrameworkFinalC13
AliBoozkurt/YFrameworkFinalC13
There isn't anything to compare.
yollinfo:master and AliBoozkurt:master are identical.

- **Choose the CP-002, and dev for destination:**

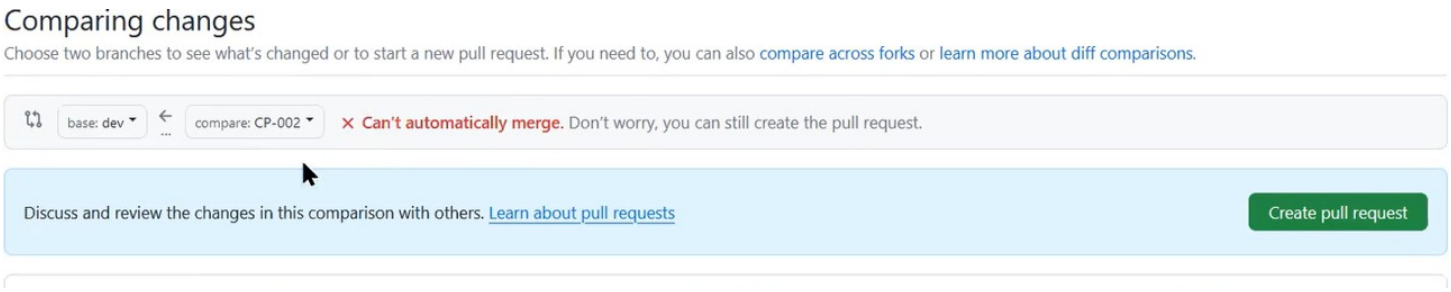

Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base: dev
compare: master
Choose a head ref
Find a branch
Branches
Tags
master
default
CP-001
CP-002
dev
There isn't anything to compare.
ll commits from master. Try switching the base for your comparison.
Showing 0 chang

Result: CP-002 Cannot be automatically merged with dev (The line: 16 on green color)


Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base: dev
compare: CP-002
Can't automatically merge. Don't worry, you can still create the pull request.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request
1 commit
1 file changed
1 contributor
Commits on Feb 11, 2024
I updated User Management Feature file with adding username and password
Ali B committed 3 minutes ago
bf0e471
Showing 1 changed file with 1 addition and 0 deletions.
Split
Unified
src/test/resources/ui_features/user-management.feature
@@ -13,3 +13,4 @@ Feature: User Management Feature
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on login page

The conflict needs to be handled manually with IntelliJ, it is not recommended by GitHub !

Do not click on : Create pull request NOBODY LIKES THIS WAY TO RESOLVE THE CONFLICT !


Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base: dev
compare: CP-002
Can't automatically merge. Don't worry, you can still create the pull request.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request

Resolving Conflicts

- When conflict occurs, GIT is nice enough to mark the problematic area in the file by enclosing it in "<<<<<<< HEAD" and ">>>>>>> [other/branch/name]".

```
1  <<<<<<< HEAD
2  This line was committed while working in the "login-box" branch.
3  =======
4  This line, in contrast, was committed while working in the "contact-form" branch.
5  >>>>>>> refs/heads/contact-form
```

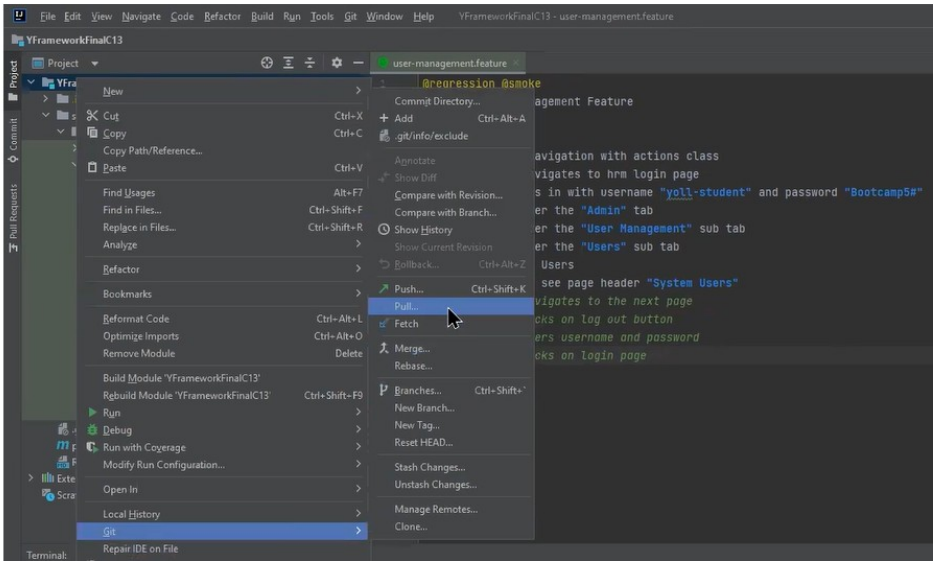
- Our job is now to clean up these lines: when we're done, the file should look exactly as we want it to look.
  a. It can be necessary to consult the teammate who wrote the conflicting changes to decide which code is finally correct.
  b. Maybe your version needs to override theirs.
  c. Maybe their version needs to override yours.
  d. Or maybe you need to keep both changes and modify some of it. (most cases)

In order to resolve the previous conflict in IntelliJ, Go to CP-002:

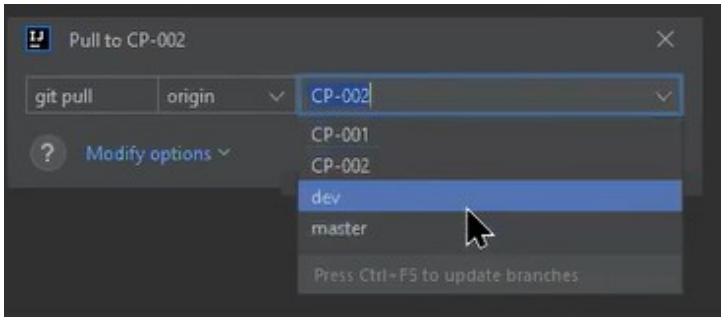
- **Checkout : CP-002.**
- **Go to Project / Right Click on it / Git / Pull.**
- **Choose dev, Then Pull to CP-002.**
- **Merge. (we can see the : <<<<<<<< HEAD) : It is what I want to change(Adding :#And user clicks on Login page) And under it, there is what is already on GitHub (#And user clicks on homepage).**
- **Make a change, for example: accept the LEFT (#And user clicks on Login page) and add it to the Result. (See the picture below)**
- **Cancel the RIGHT: (Adding :#And user clicks on homepage). (See the picture below)**
- **Apply.**
- **Execute the command: git push.**
- **See on GitHub that CP-002 is able to merge after it was (can't automatically merge). Refresh the page First. (You can Pull Request Now)**
- **Click on Create pull request, and follow the steps.**
- **Click on Merge pull request / Confirm Merge.**

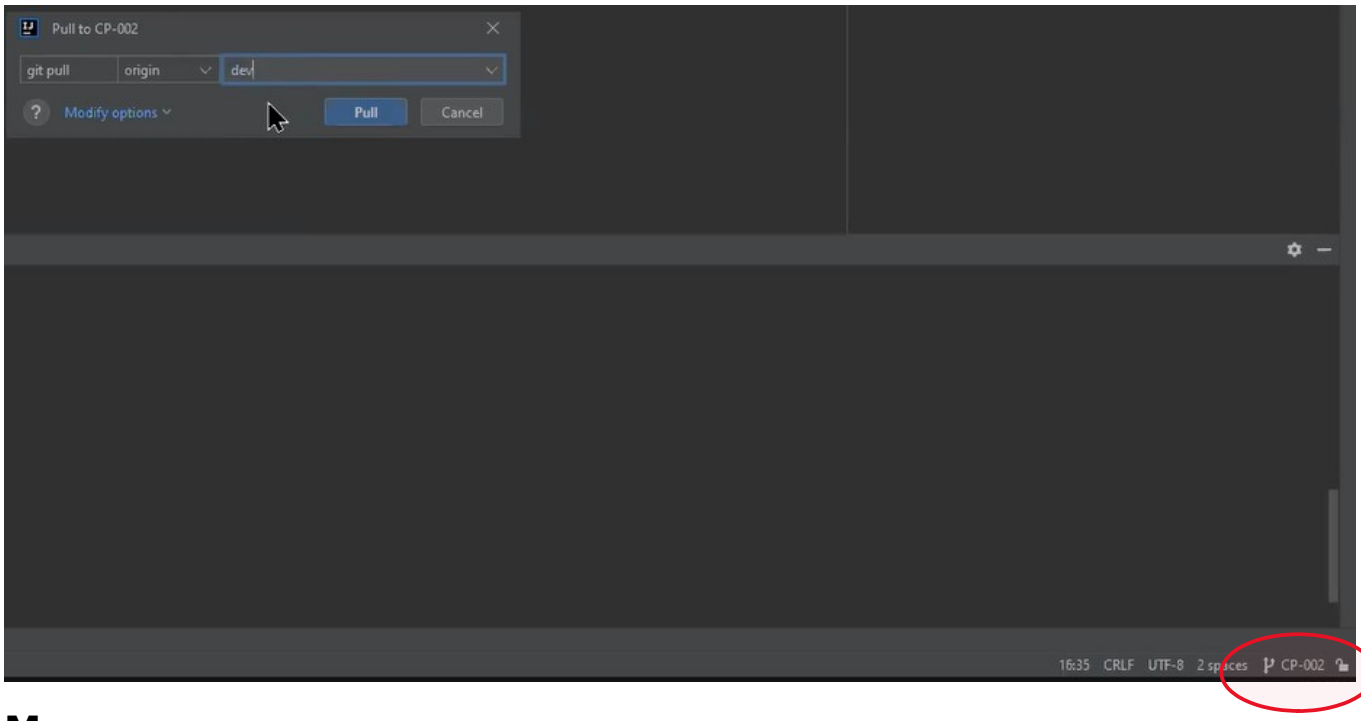
- **Check dev on GitHub to see the changes and resolving conflicts.**

- **Checkout : CP-002.**
- **Go to Project / Right Click on it / Git / Pull.**

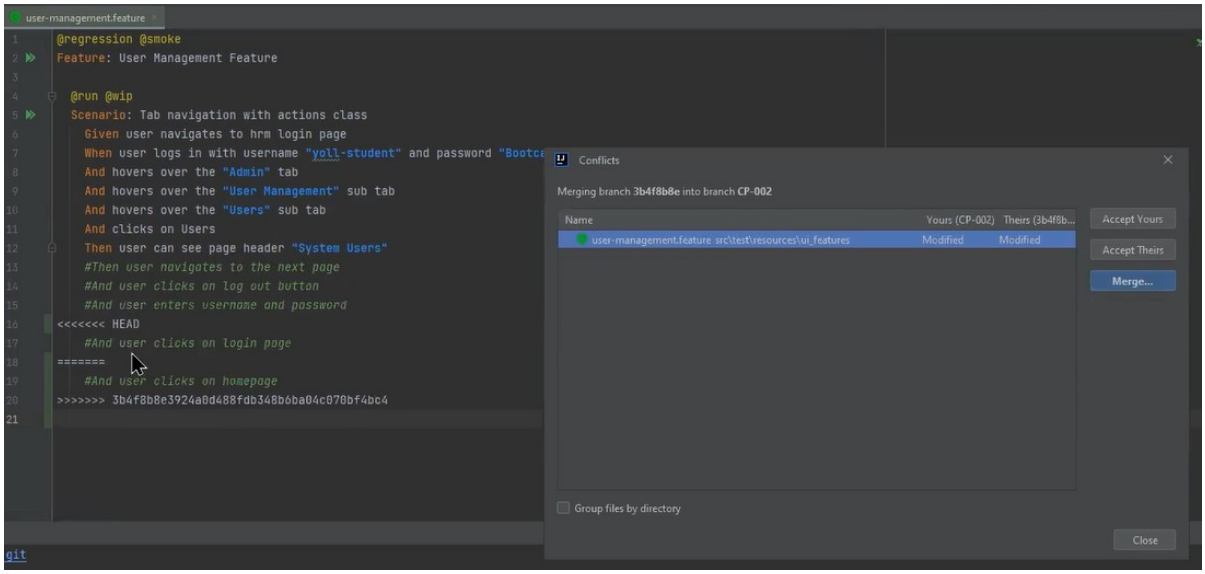


- **Choose dev, Then Pull to CP-002.**





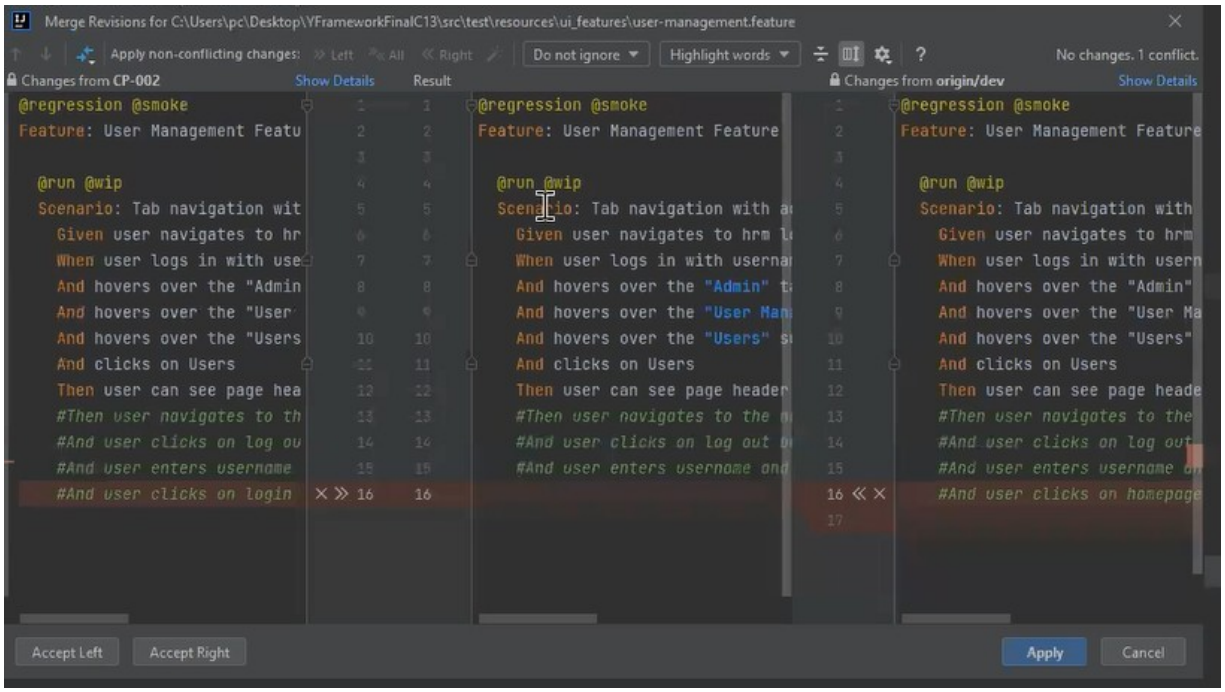
Merge.



Left: Changes from CP-002, it contains: **#And user clicks on Login page**

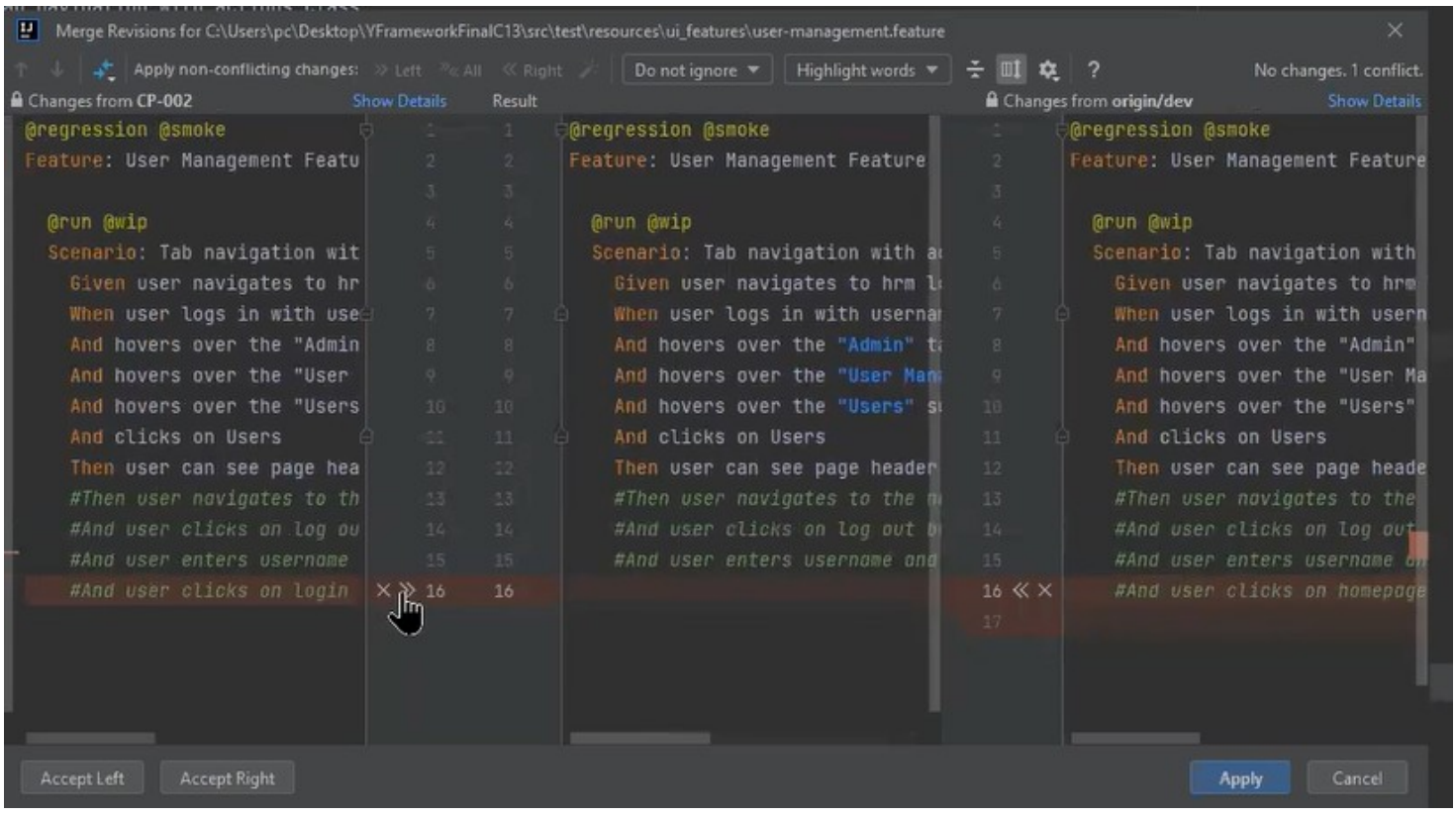
Middle: Result after conflict: Nothing because of conflict.

Dev: Changes from CP-001: **#And user clicks on homepage**

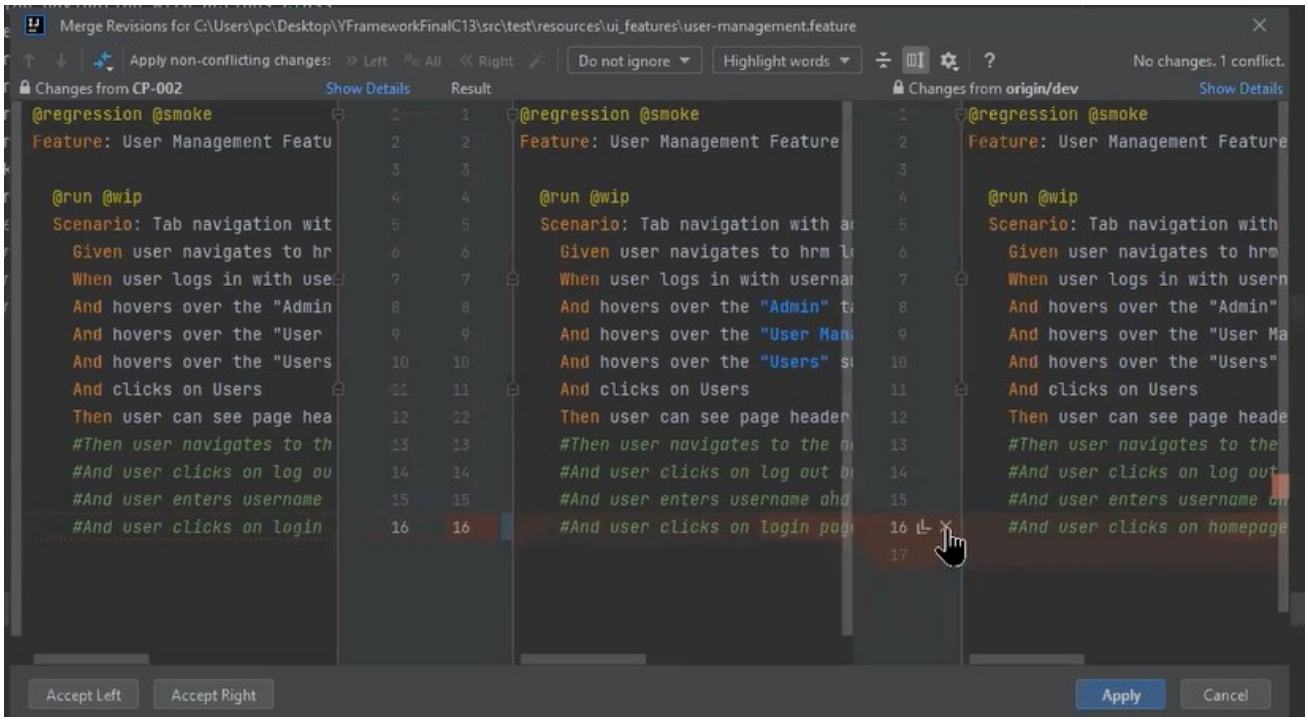


Solution:

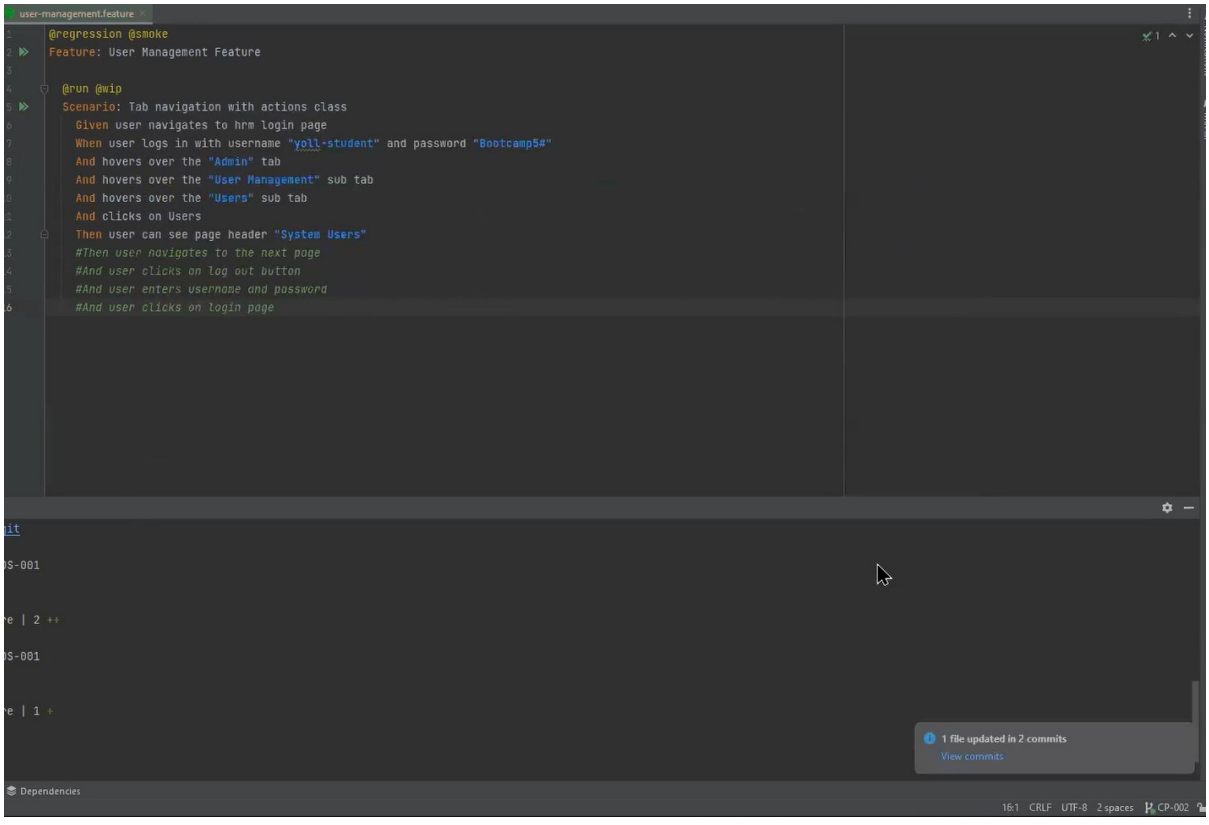
In case of accepting to change from CP-002 and adding: **#And user clicks on Login page to the RESULT (Middle)**



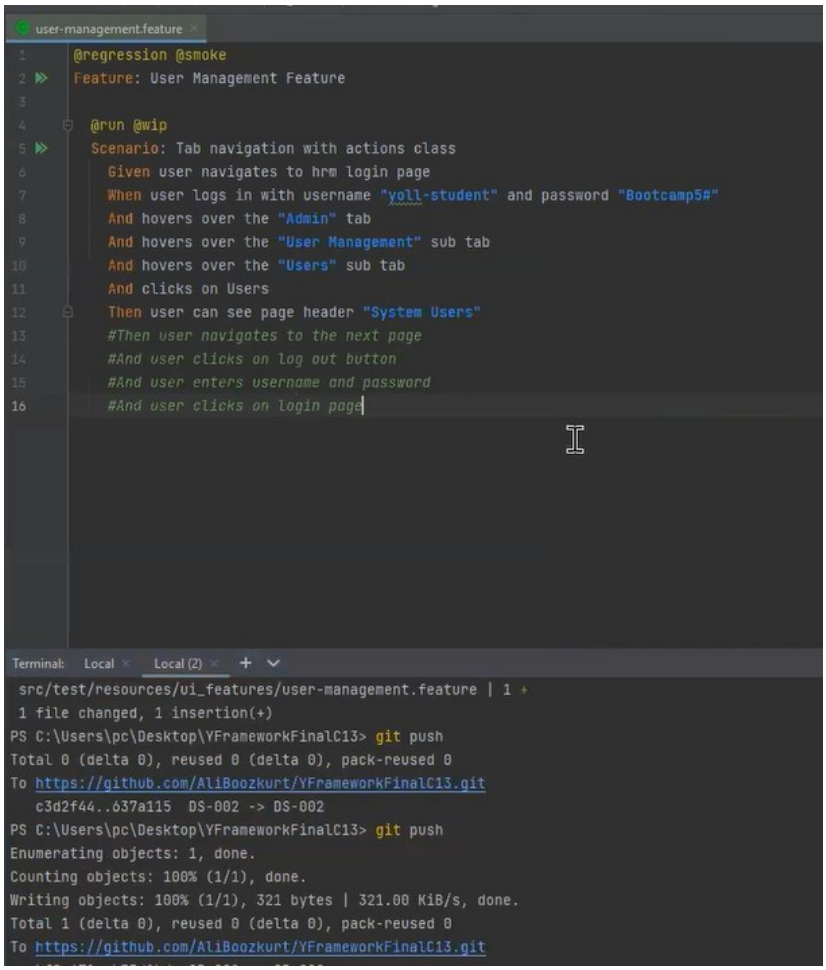
Cancelling the Right (Line 16) : #And user clicks on homepage



On CP-002, we can see the changes: **#And user clicks on Login page**


user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll+student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on login page
1 file updated in 2 commits
View commits
Dependencies
16:1 CRLF UTF-8 2 spaces CP-002

Using : git push

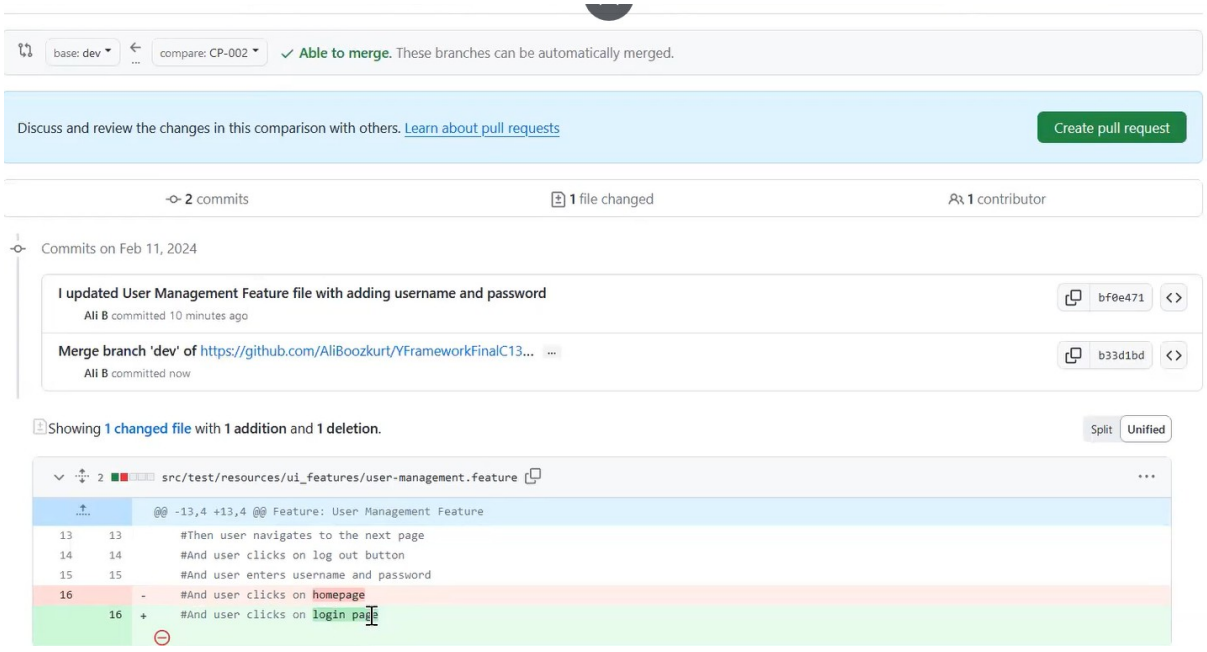

user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on login page
Terminal: Local Local (2)
src/test/resources/ui_features/user-management.feature | 1 +
1 file changed, 1 insertion(+)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AliBoozkurt/YFrameworkFinalC13.git
c3d2f44..637a115 DS-002 -> DS-002
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git push
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 321 bytes | 321.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AliBoozkurt/YFrameworkFinalC13.git

Now, we can see that CP-002 is ABLE TO MERGE on GitHub:

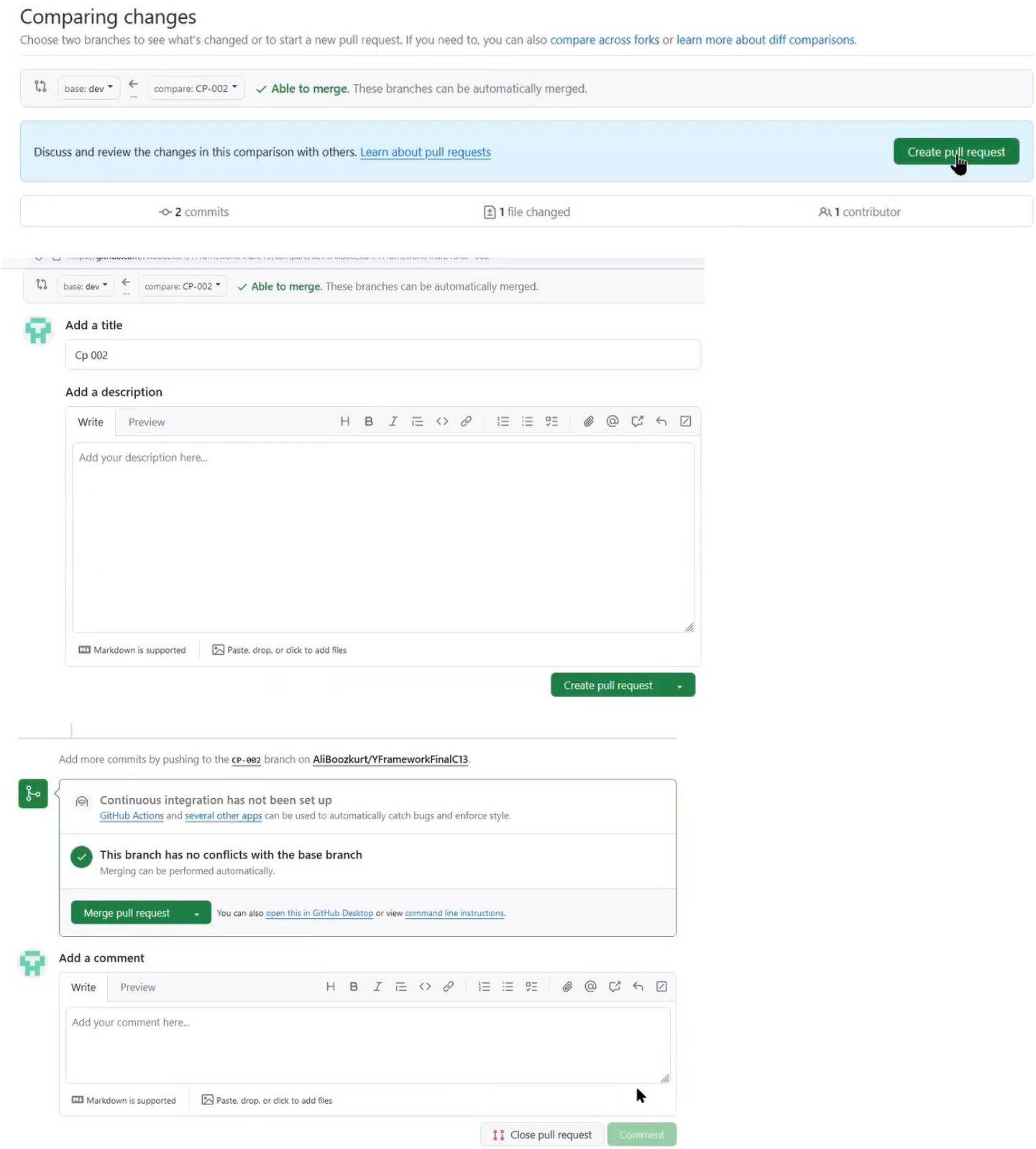
Refresh the page first !

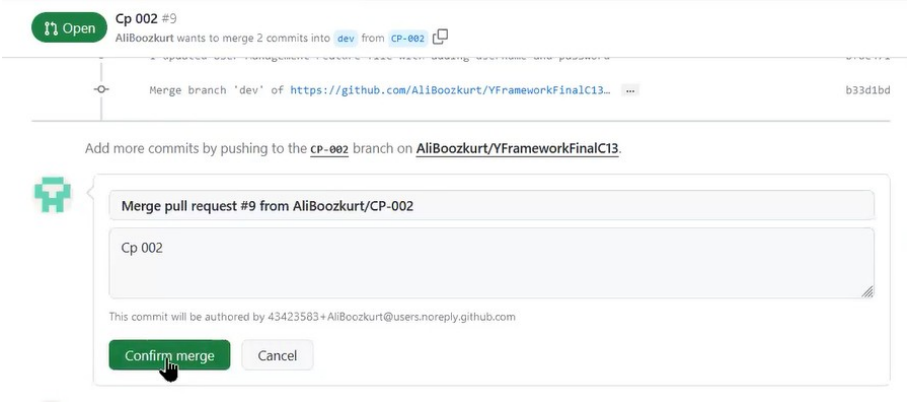
The red color is the line ignored.

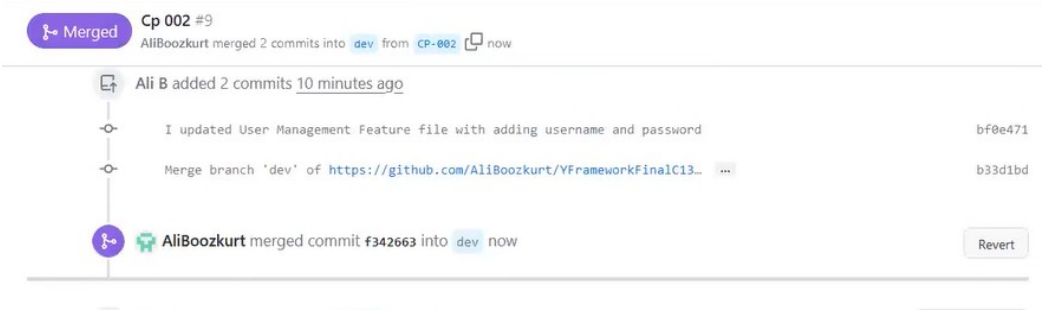
The green color is the line accepted.


base: dev
compare: CP-002
Able to merge. These branches can be automatically merged.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request
2 commits
1 file changed
1 contributor
Commits on Feb 11, 2024
I updated User Management Feature file with adding username and password
Ali B committed 10 minutes ago
bf0e471
Merge branch 'dev' of https://github.com/AliBoozkurt/YFrameworkFinalC13...
Ali B committed now
b33d1bd
Showing 1 changed file with 1 addition and 1 deletion.
Split
Unified
src/test/resources/ui_features/user-management.feature
@@ -13,4 +13,4 @@ Feature: User Management Feature
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage
#And user clicks on login page

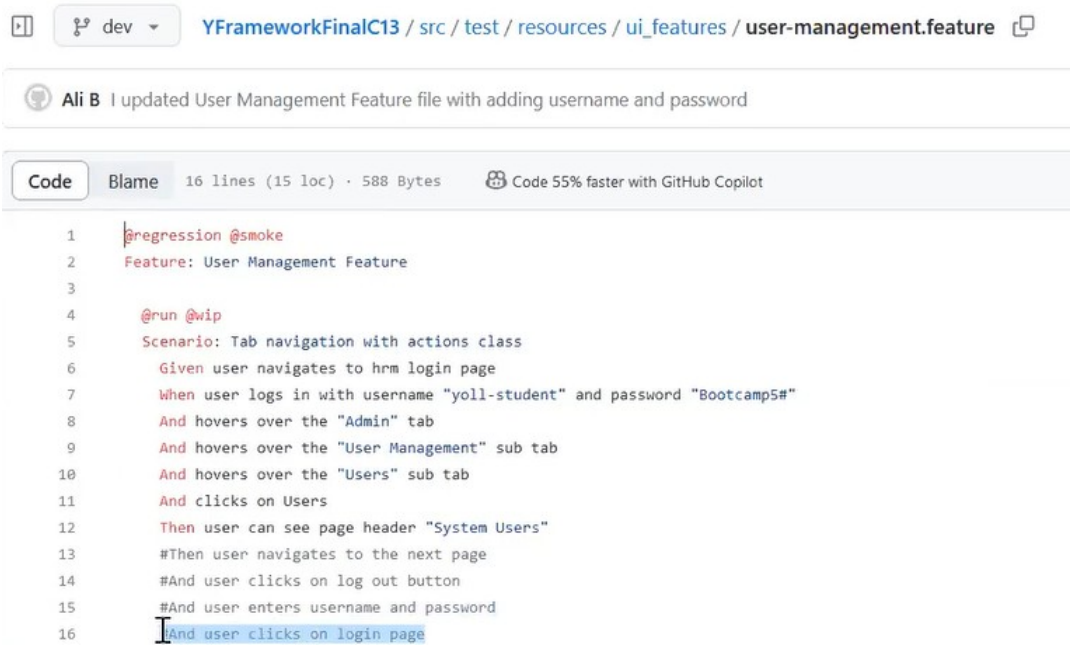
Create pull request


Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base: dev
compare: CP-002
Able to merge. These branches can be automatically merged.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request
2 commits
1 file changed
1 contributor
base: dev
compare: CP-002
Able to merge. These branches can be automatically merged.
Add a title
Cp 002
Add a description
Write
Preview
Add your description here...
Markdown is supported
Paste, drop, or click to add files
Create pull request
Add more commits by pushing to the CP-002 branch on AliBoozkurt/YFrameworkFinalC13.
Continuous integration has not been set up
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.
This branch has no conflicts with the base branch
Merging can be performed automatically.
Merge pull request
You can also open this in GitHub Desktop or view command line instructions.
Add a comment
Write
Preview
Add your comment here...
Markdown is supported
Paste, drop, or click to add files
Close pull request
Comment


Open
Cp 002 #9
AliBoozkurt wants to merge 2 commits into dev from CP-002
Merge branch 'dev' of https://github.com/AliBoozkurt/YFrameworkFinalC13…
b33d1bd
Add more commits by pushing to the CP-002 branch on AliBoozkurt/YFrameworkFinalC13.
Merge pull request #9 from AliBoozkurt/CP-002
Cp 002
This commit will be authored by 43423583+AliBoozkurt@users.noreply.github.com
Confirm merge
Cancel


Merged
Cp 002 #9
AliBoozkurt merged 2 commits into dev from CP-002 now
Ali B added 2 commits 10 minutes ago
I updated User Management Feature file with adding username and password
bf0e471
Merge branch 'dev' of https://github.com/AliBoozkurt/YFrameworkFinalC13…
b33d1bd
AliBoozkurt merged commit f342663 into dev now
Revert

Check dev on GitHub to see the changes and resolving conflicts.


dev
YFrameworkFinalC13 / src / test / resources / ui_features / user-management.feature
Ali B I updated User Management Feature file with adding username and password
Code
Blame
16 lines (15 loc) · 588 Bytes
Code 55% faster with GitHub Copilot
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on login page

TASK

CREATE NEW BRANCHES on GtiHub:

AB-1 from dev

AB-2 from dev

Then update AB-1:

By adding the line 15:

#And TESTER enters PASSWORD and USERNAME

And by adding the lines 17, 18, 19:

#And user navigates to search page

#And user searches for “Book”

#And user selects the first item from the list ***Use Commit and Push***

Then update AB-2:

By adding the line 15:

#And TESTER enters PASSWORD and USERNAME

And by adding the lines 17, 18, 19, 20:

#And user navigates to search page

#And user searches for “Book”

#And user selects the first item from the list

#And user logs out from all accounts ***Use: Commit and Push button***

Click on pull requests / New Pull request / Choose Repository /

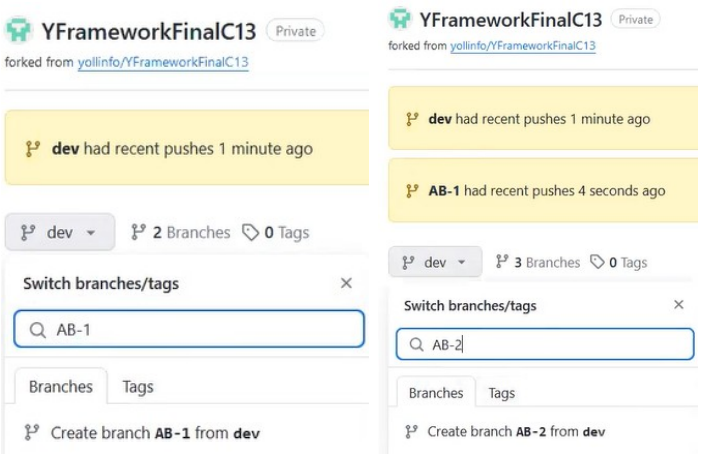
Select the branches : (AB-1 TO dev AND AB-2 TO dev), FOR BOTH OF THEM (AB-1 AND AB-2):

For AB-1 Create pull request / Merge pull request/ Confirm merge

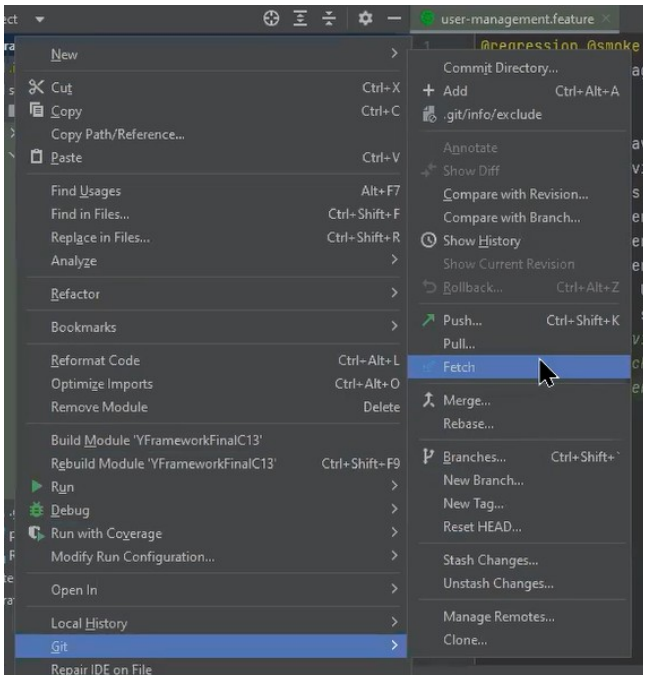
For AB-2 Create pull request /When you see the conflict>> Pull from IntelliJ / Merge/ Then you can see changes<<<<<<<HEAD.

Decide which one to add Right or Left to the middle. Then Apply. Use the command : git push.

Refresh GitHub, Create pull request



Fetch dev to download changes from a remote repository on GitHub without automatically merging them into your local branch.



Then update AB-1

By adding the line 15:

#And TESTER enters PASSWORD and USERNAME

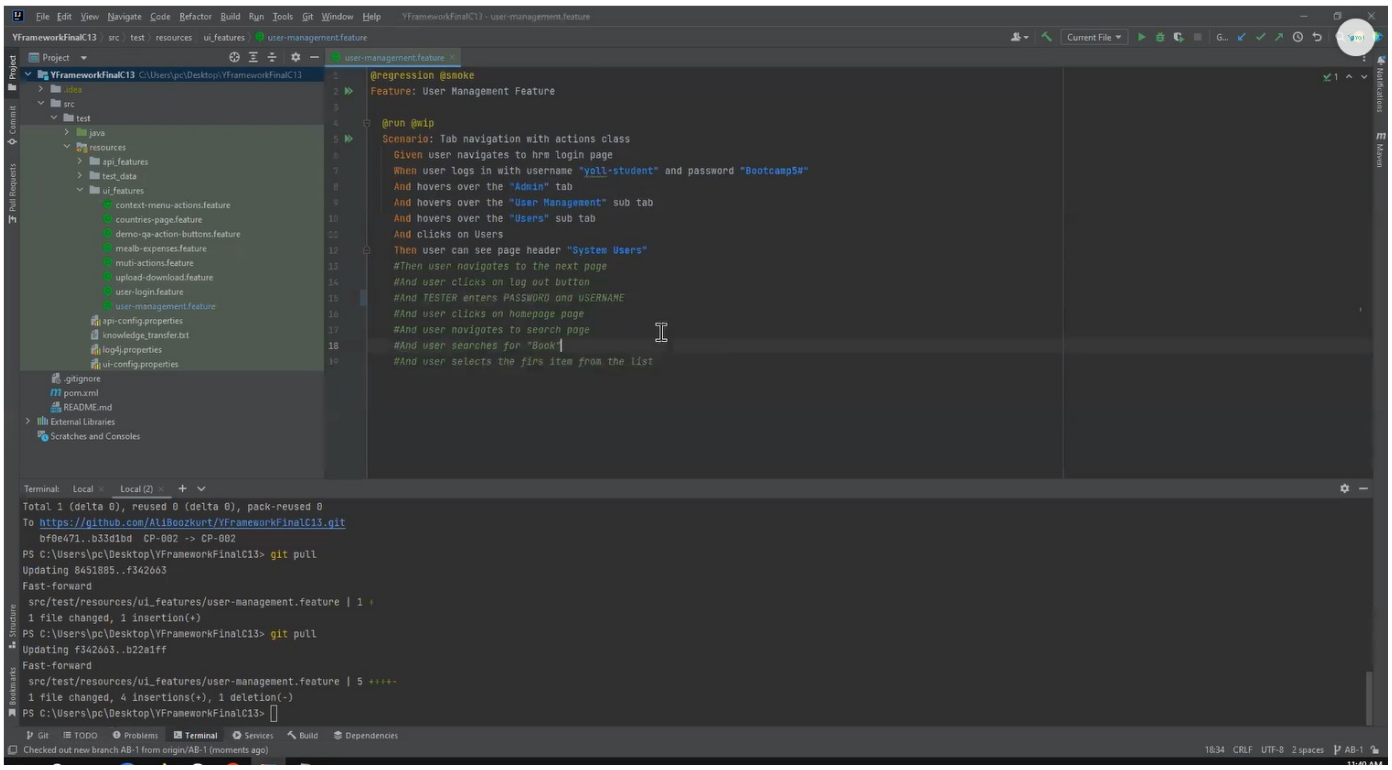
And by adding the lines 17, 18, 19:

#And user navigates to search page

#And user searches for “Book”

#And user selects the first item from the list

Use Commit and Push


@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And TESTER enters PASSWORD and USERNAME
#And user clicks on homepage page
#And user navigates to search page
#And user searches for "Book"
#And user selects the firs item from the list
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AliBozkurt/YFrameworkFinalC13.git
bf0e471..b33d1bd CP-002 -> CP-002
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git pull
Updating 8451885..f342663
Fast-forward
src/test/resources/ui_features/user-management.feature | 1 +
1 file changed, 1 insertion(+)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git pull
Updating f342663..b22a1ff
Fast-forward
src/test/resources/ui_features/user-management.feature | 5 ++++-
1 file changed, 4 insertions(+), 1 deletion(-)
PS C:\Users\pc\Desktop\YFrameworkFinalC13>

Then update AB-2

By adding the line 15:

#And TESTER enters PASSWORD and USERNAME

And by adding the lines 17, 18, 19, 20:

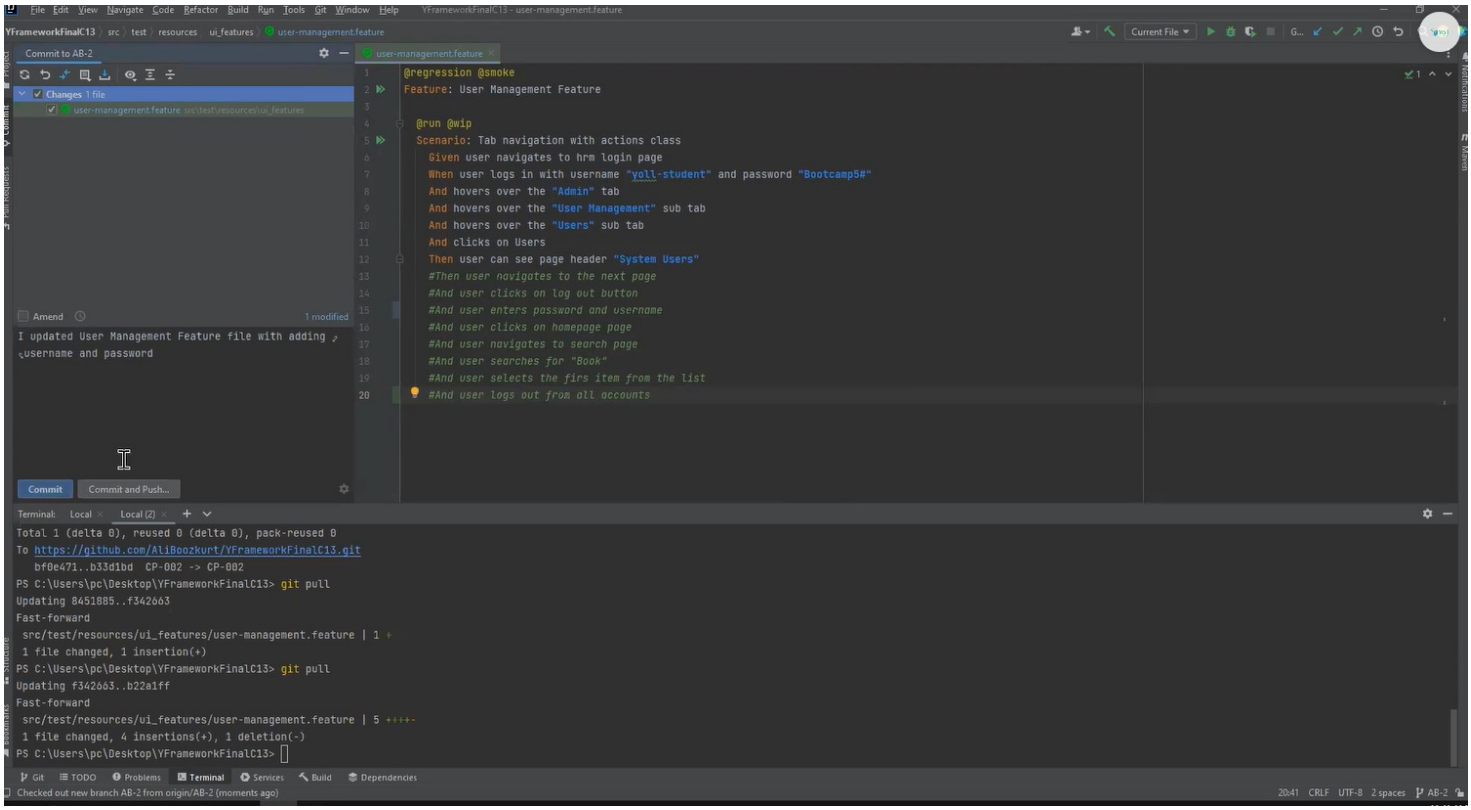
#And user navigates to search page

#And user searches for “Book”

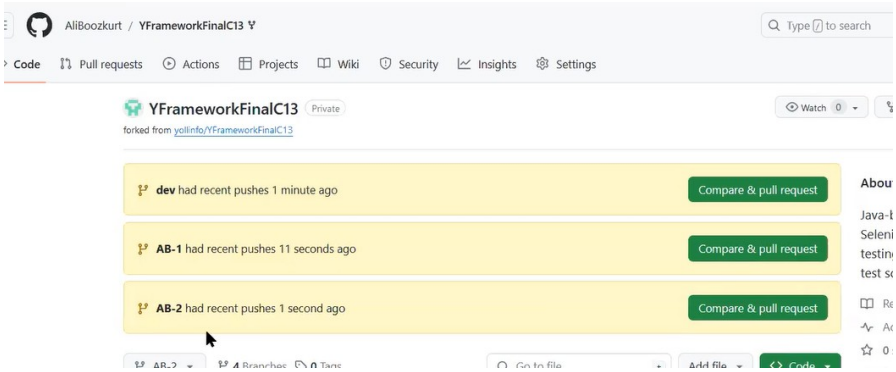
#And user selects the first item from the list

#And user logs out from all accounts

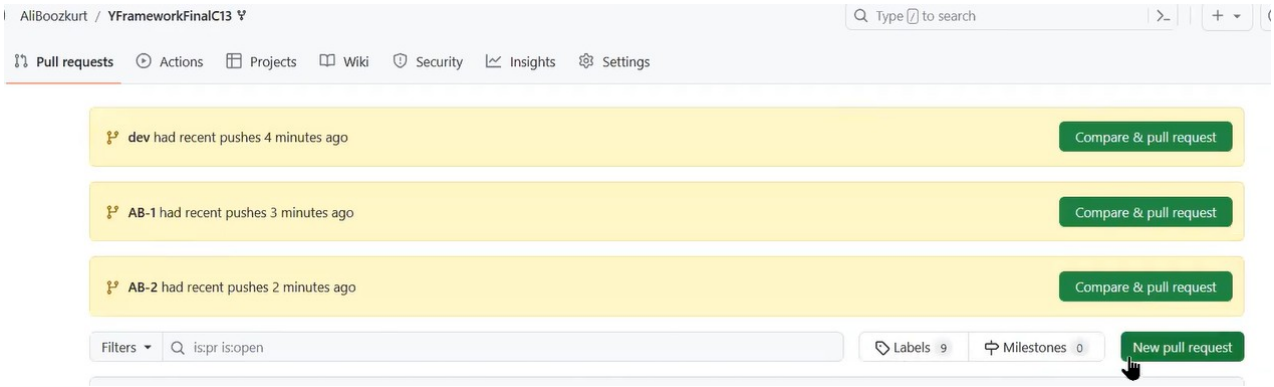
Use: Commit and Push button


@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters password and username
#And user clicks on homepage page
#And user navigates to search page
#And user searches for "Book"
#And user selects the firs item from the list
#And user logs out from all accounts
I updated User Management Feature file with adding username and password
Commit
Commit and Push...
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AliBoozkurt/YFrameworkFinalC13.git
bf0e471..b33d1bd CP-002 -> CP-002
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git pull
Updating 8451885..f342663
Fast-forward
src/test/resources/ui_features/user-management.feature | 1 +
1 file changed, 1 insertion(+)
PS C:\Users\pc\Desktop\YFrameworkFinalC13> git pull
Updating f342663..b22a1ff
Fast-forward
src/test/resources/ui_features/user-management.feature | 5 ++++-
1 file changed, 4 insertions(+), 1 deletion(-)
PS C:\Users\pc\Desktop\YFrameworkFinalC13>

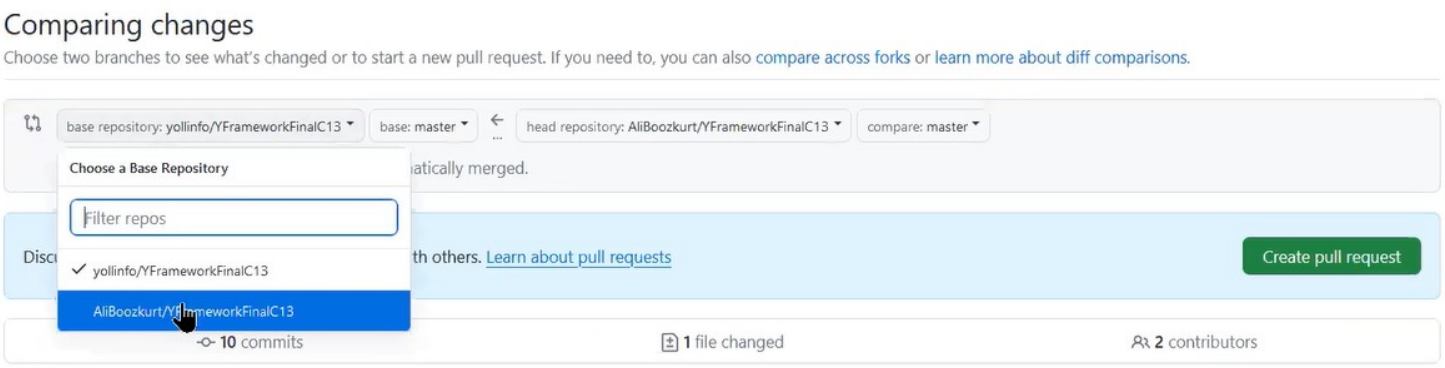
Click on pull requests



Click on New Pull requst button:

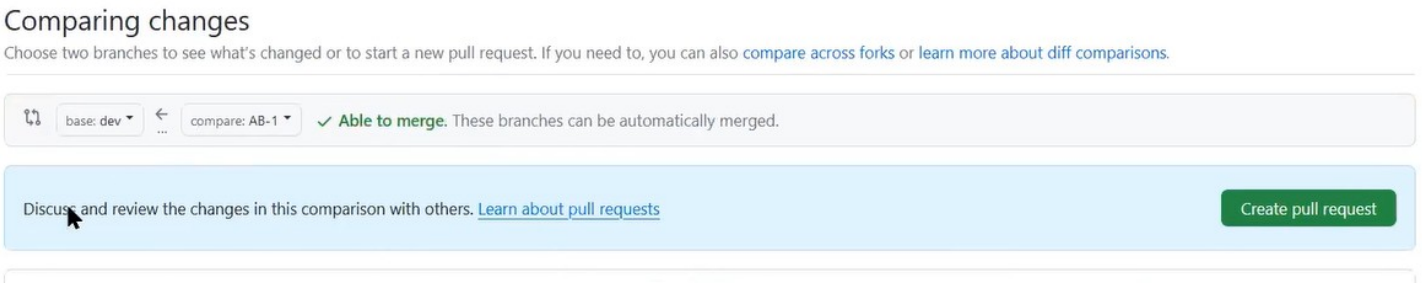


Choose Repository


Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base repository: yollinfo/YFrameworkFinalC13
base: master
head repository: AliBoozkurt/YFrameworkFinalC13
compare: master
Choose a Base Repository
Filter repos
yollinfo/YFrameworkFinalC13
Create pull request
10 commits
1 file changed
2 contributors

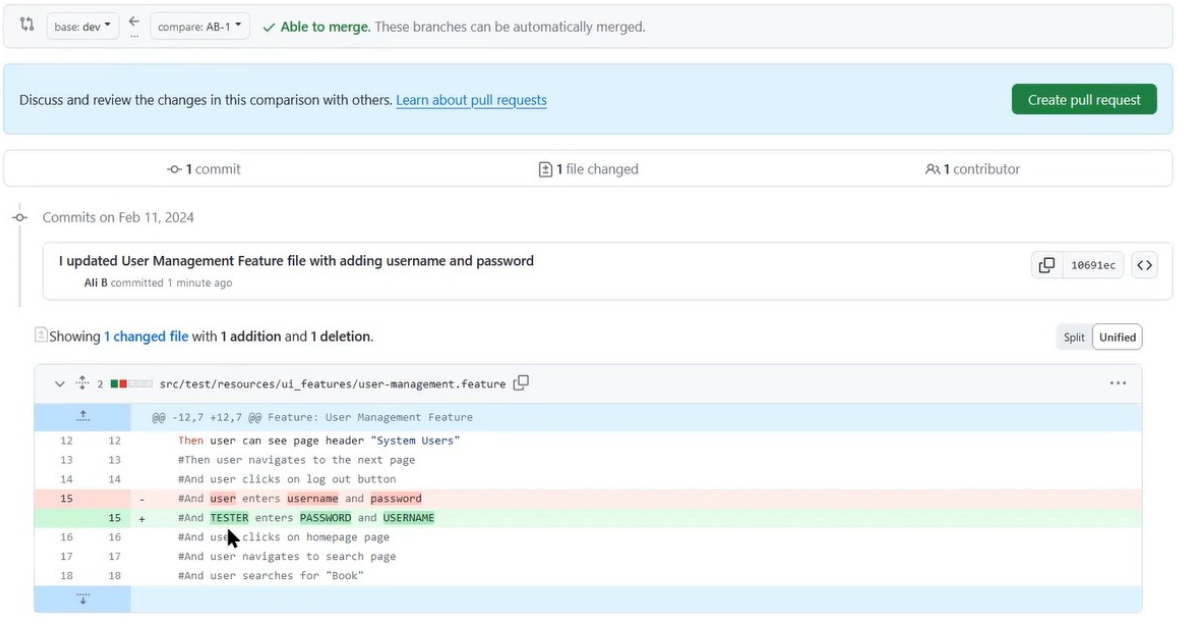
Select the branches : (AB-1 TO dev AND AB-2 TO dev)

1- AB-1 TO dev

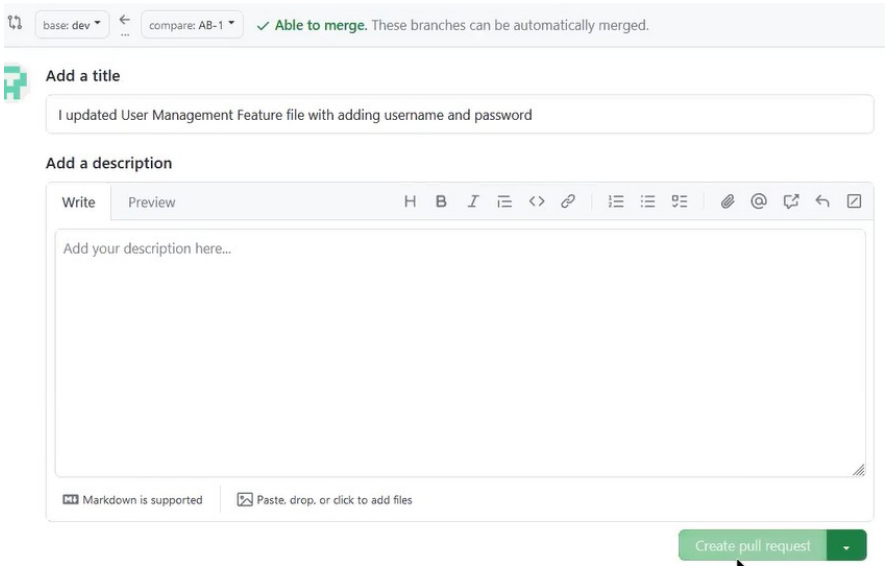

Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base: dev
compare: AB-1
Able to merge. These branches can be automatically merged.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request

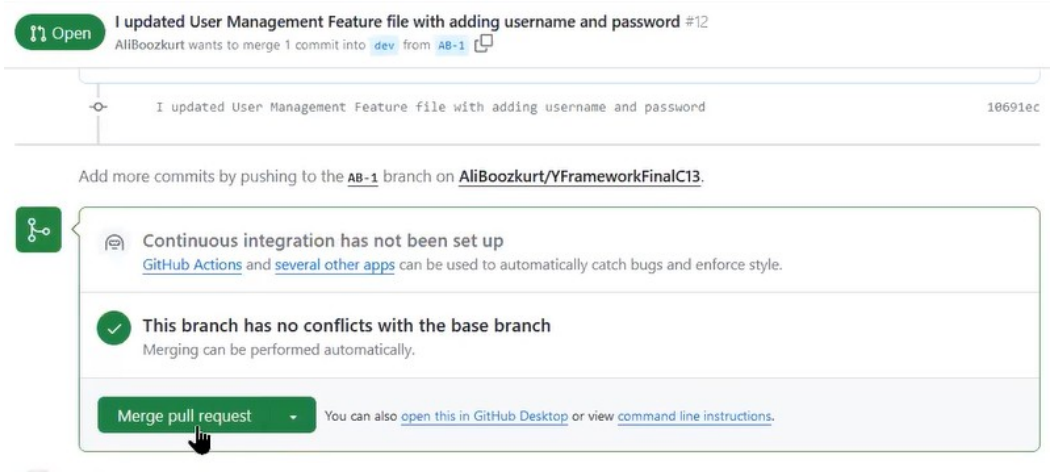
I want to change **usename** and **password** in **lower case** to **USERNAME** and **PASSWORD** in **upper case**, and change **user** to **TESTER.**

Then Click on Create pull request Button.

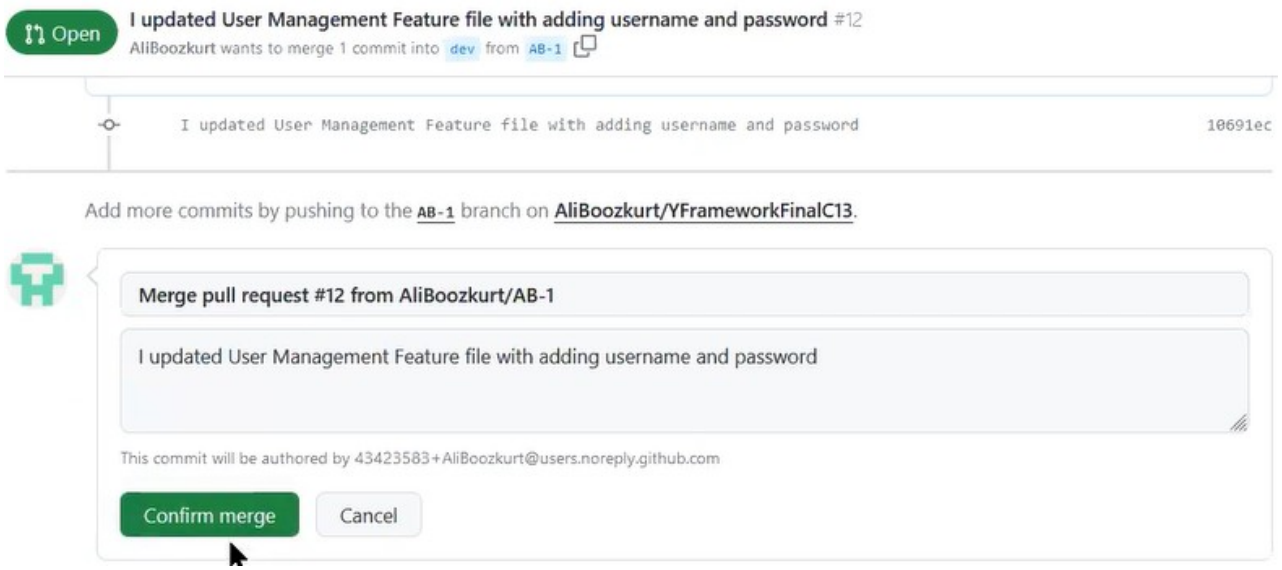

base: dev
compare: AB-1
Able to merge. These branches can be automatically merged.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request
1 commit
1 file changed
1 contributor
Commits on Feb 11, 2024
I updated User Management Feature file with adding username and password
Ali B committed 1 minute ago
10691ec
Showing 1 changed file with 1 addition and 1 deletion.
Split
Unified
src/test/resources/ui_features/user-management.feature
@@ -12,7 +12,7 @@ Feature: User Management Feature
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And TESTER enters PASSWORD and USERNAME
#And user clicks on homepage page
#And user navigates to search page
#And user searches for "Book"

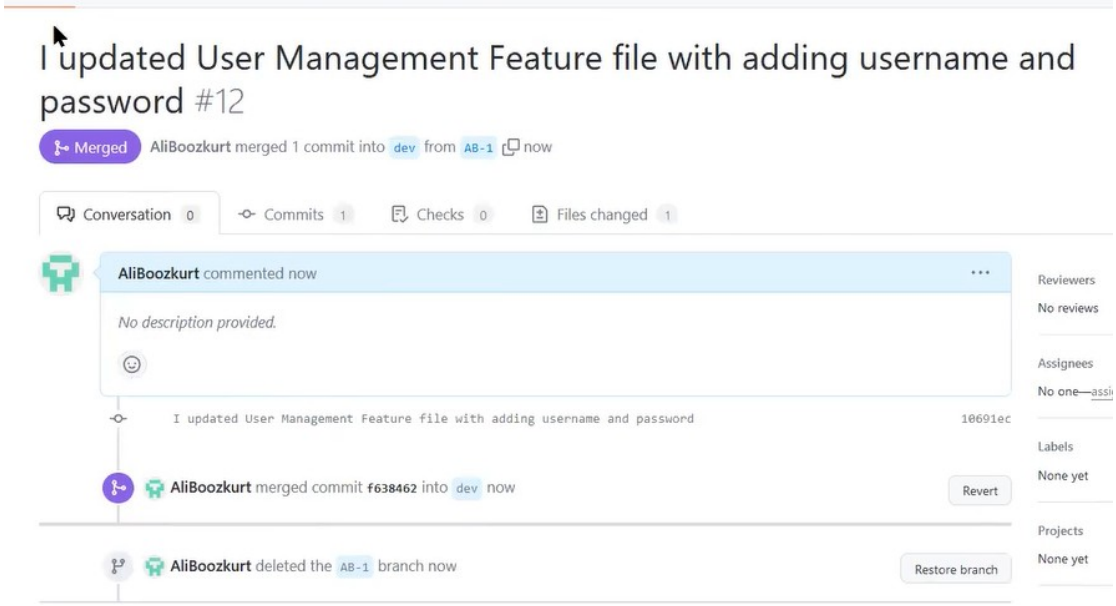
Follow the steps:


base: dev
compare: AB-1
Able to merge. These branches can be automatically merged.
Add a title
I updated User Management Feature file with adding username and password
Add a description
Write
Preview
Add your description here...
Markdown is supported
Paste, drop, or click to add files
Create pull request


Open
I updated User Management Feature file with adding username and password #12
AliBoozkurt wants to merge 1 commit into dev from AB-1
I updated User Management Feature file with adding username and password
10691ec
Add more commits by pushing to the AB-1 branch on AliBoozkurt/YFrameworkFinalC13.
Continuous integration has not been set up
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.
This branch has no conflicts with the base branch
Merging can be performed automatically.
Merge pull request
You can also open this in GitHub Desktop or view command line instructions.

Confirm merge:


Open
I updated User Management Feature file with adding username and password #12
AliBoozkurt wants to merge 1 commit into dev from AB-1
I updated User Management Feature file with adding username and password
10691ec
Add more commits by pushing to the AB-1 branch on AliBoozkurt/YFrameworkFinalC13.
Merge pull request #12 from AliBoozkurt/AB-1
I updated User Management Feature file with adding username and password
This commit will be authored by 43423583+AliBoozkurt@users.noreply.github.com
Confirm merge
Cancel



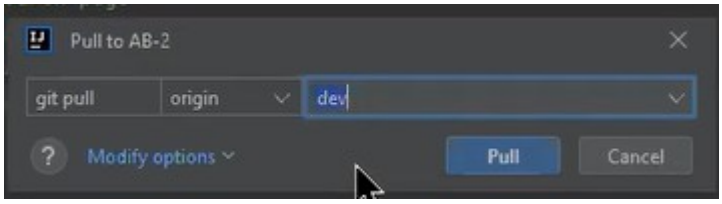
2- AB-1 TO dev:

We can see a merge conflict,We are unable to merge AB-2 to dev

We must pull AB-2 to dev first !

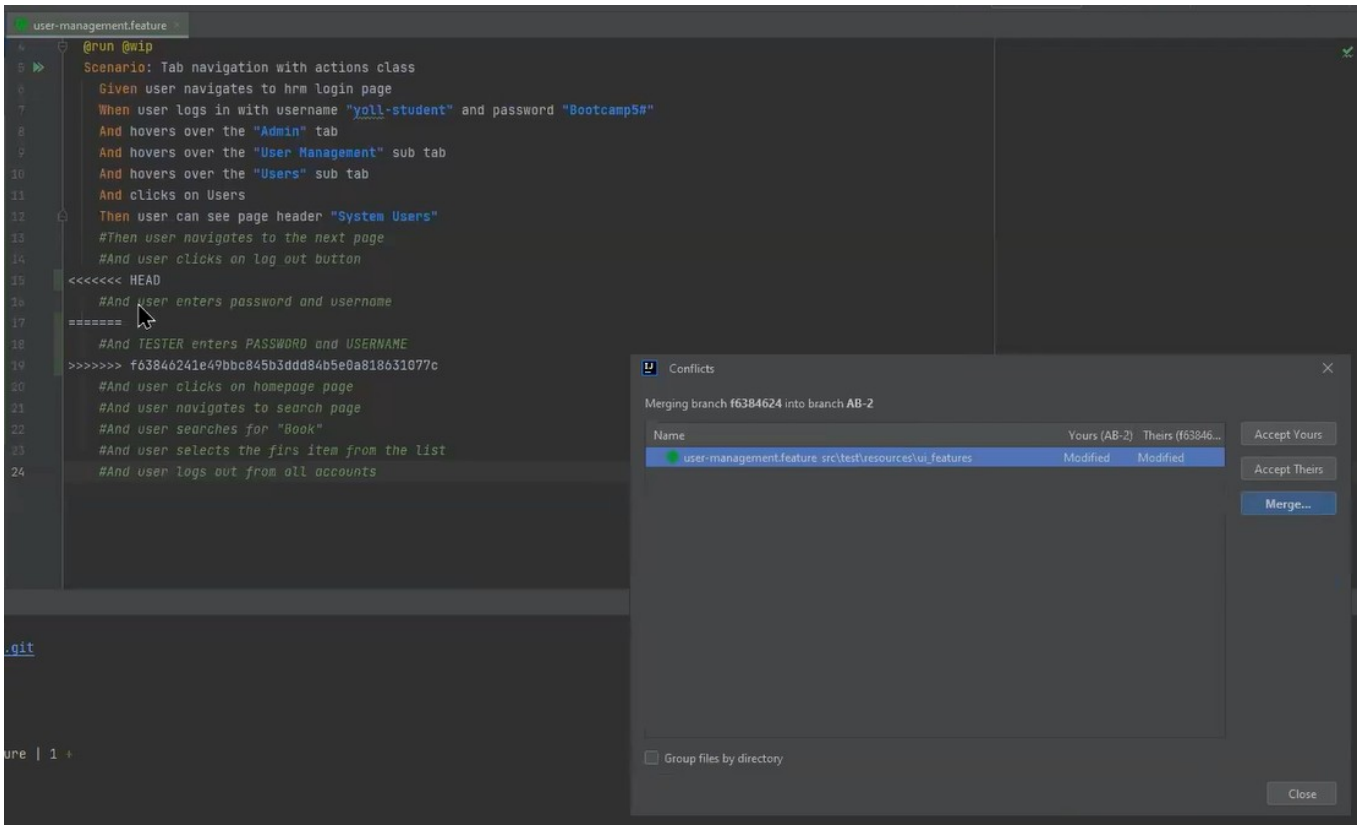
>> Checkout AB-2

>>Git/Pull

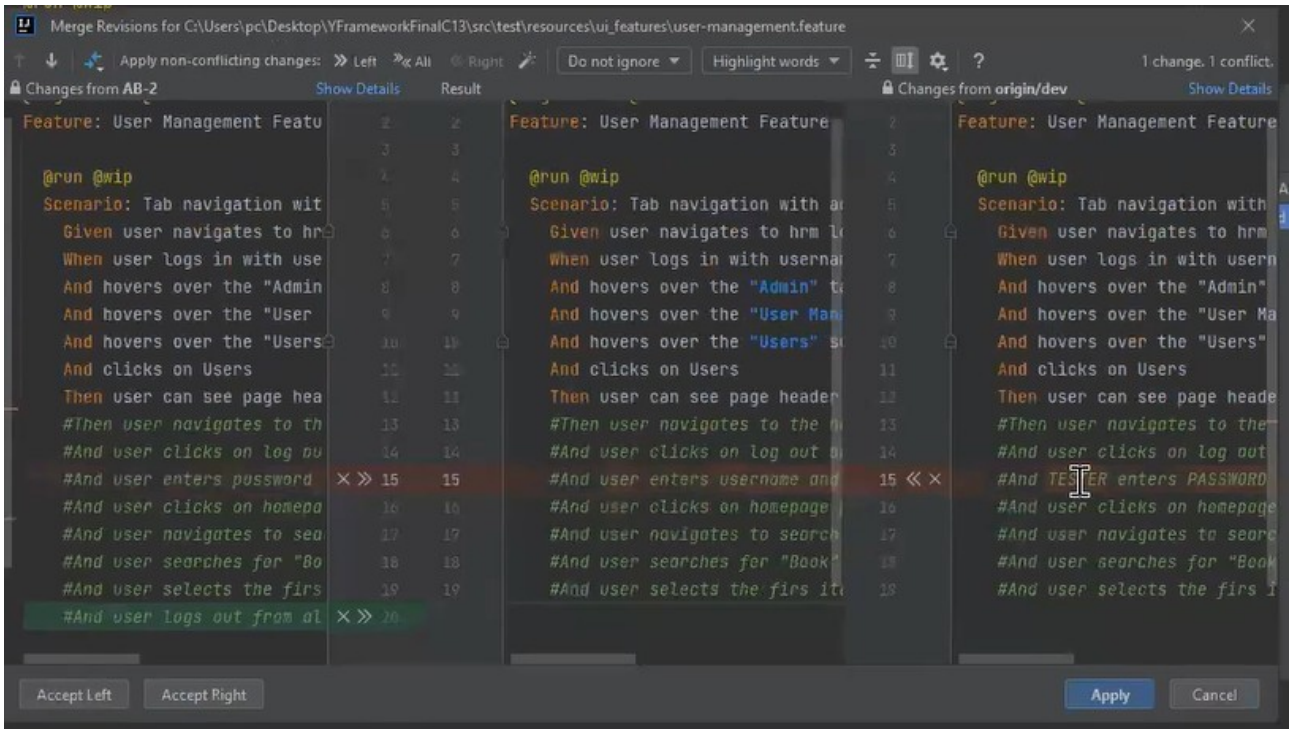


We can see the changes now:

<<<<<<<<<<HEAD

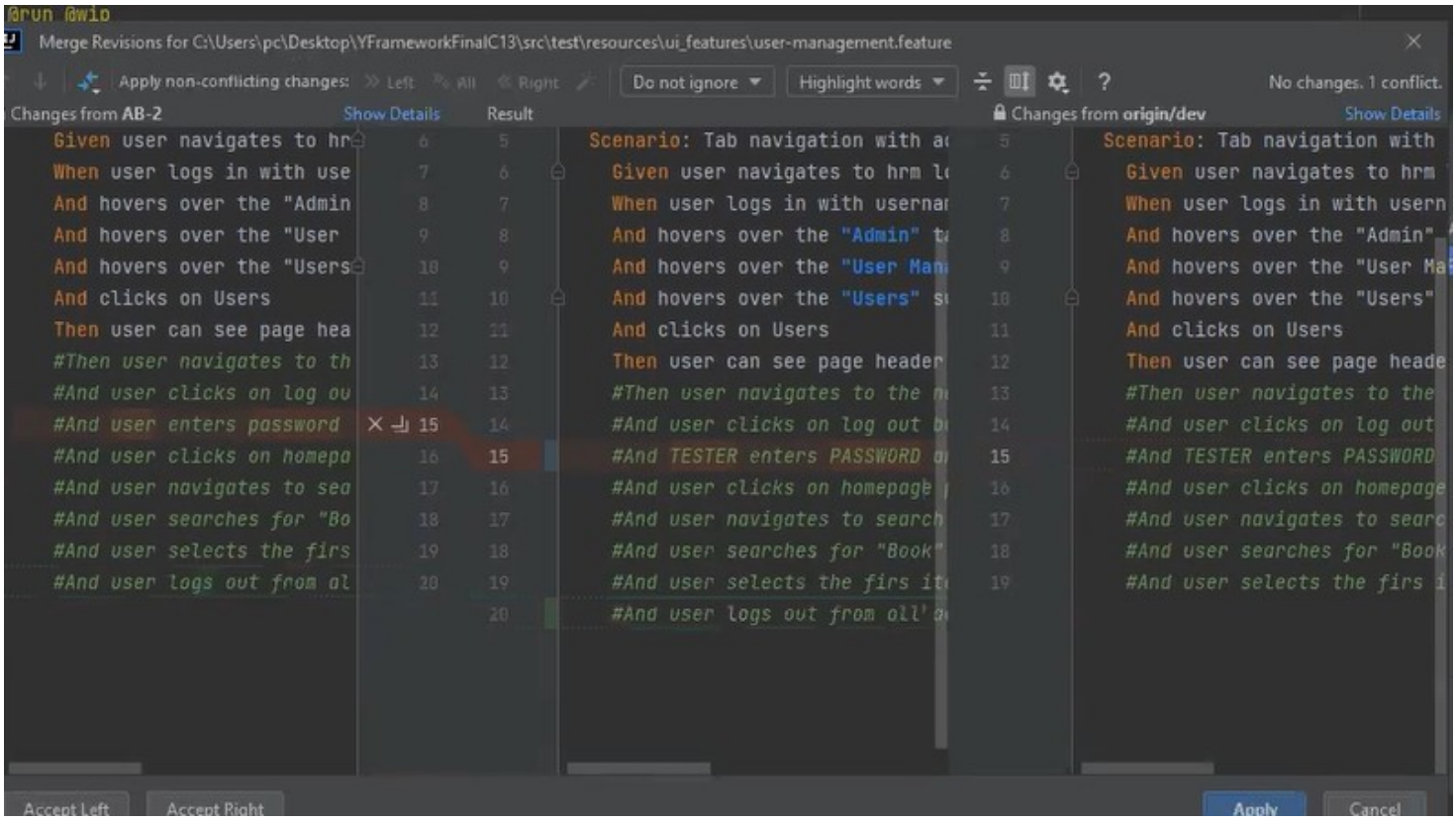

user-management.feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
<<<<<<< HEAD
#And user enters password and username
=======
#And TESTER enters PASSWORD and USERNAME
>>>>>>> f63846241e49bbc845b3ddd84b5e0a818631077c
#And user clicks on homepage page
#And user navigates to search page
#And user searches for "Book"
#And user selects the firs item from the list
#And user logs out from all accounts
Conflicts
Merging branch f6384624 into branch AB-2
Name
Yours (AB-2)
Theirs (f63846...
user-management.feature src\test\resources\ui_features
Modified
Modified
Accept Yours
Accept Theirs
Merge...
Group files by directory
Close

Decide which one to add Right or Left to the middle.

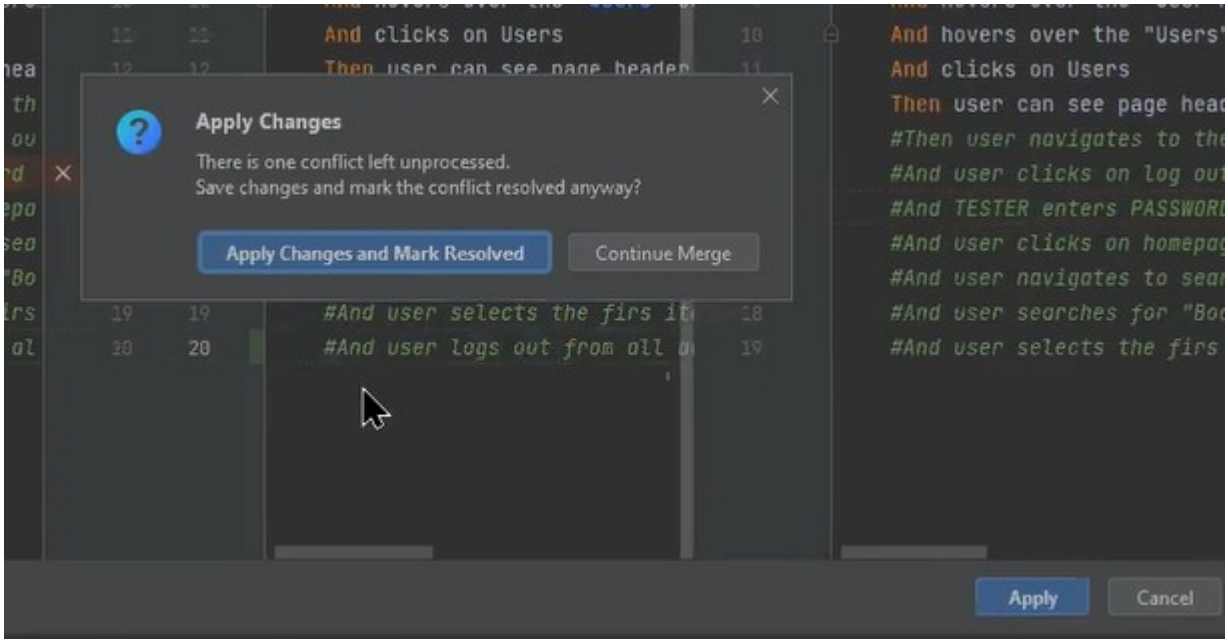


I decide to put *#And TESTER enters PASSWORD and USERNAME* FROM THE LEFT

AND *#And user logs out from all accounts* FROM THE RIGHT



Apply



Use the command : git push

Create pull request

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.

base: dev ← compare: AB-2 ✕ **Can't automatically merge.** Don't worry, you can still create the pull request.

Discuss and review the changes in this comparison with others. Learn about pull requests

Create pull request

1 commit | **1** file changed | **1** contributor

Commits on Feb 11, 2024

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks. Learn more about diff comparisons here.

base: dev ← compare: AB-2 ✓ **Able to merge.** These branches can be automatically merged.

Add a title

Ab 2

Add a description

Write | Preview

Add your description here...

Reviewers

No reviews—at least 0 approving reviews are required.

Assignees

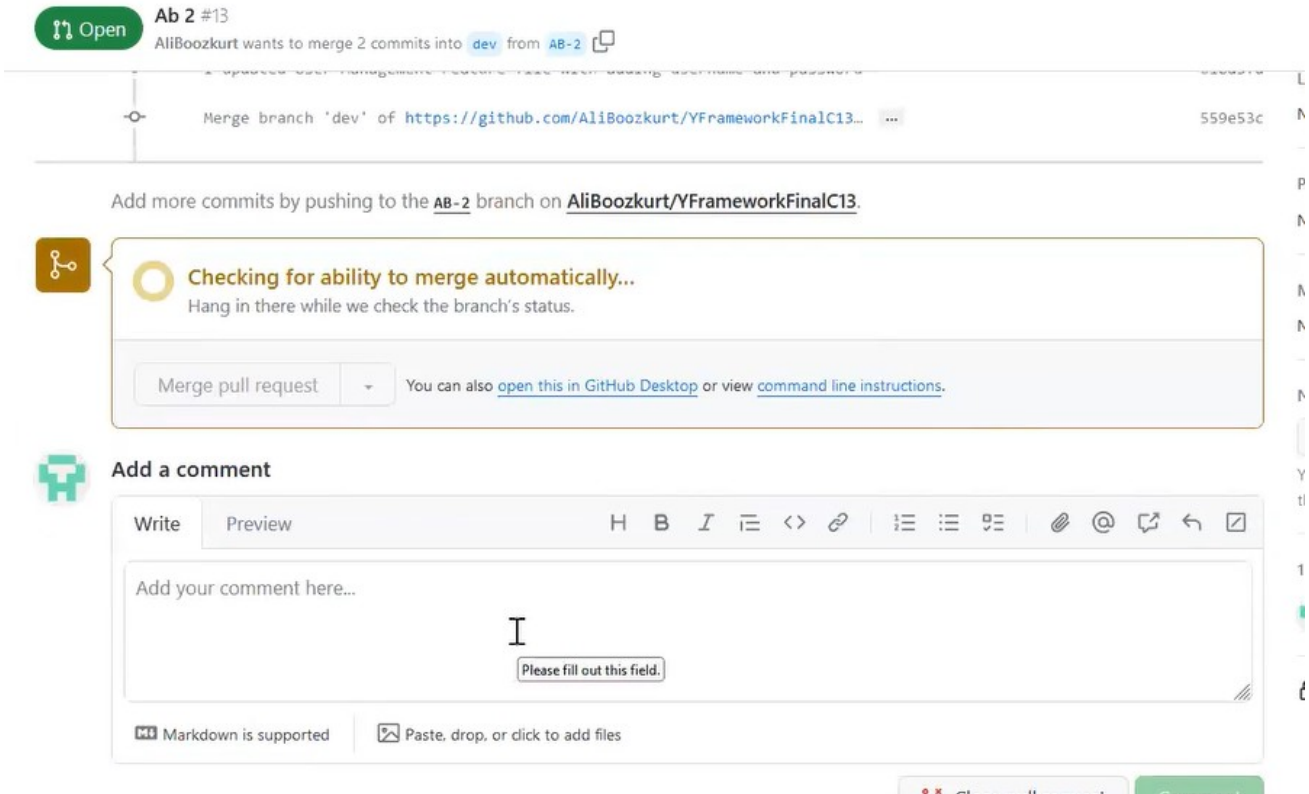
No one—assign yourself

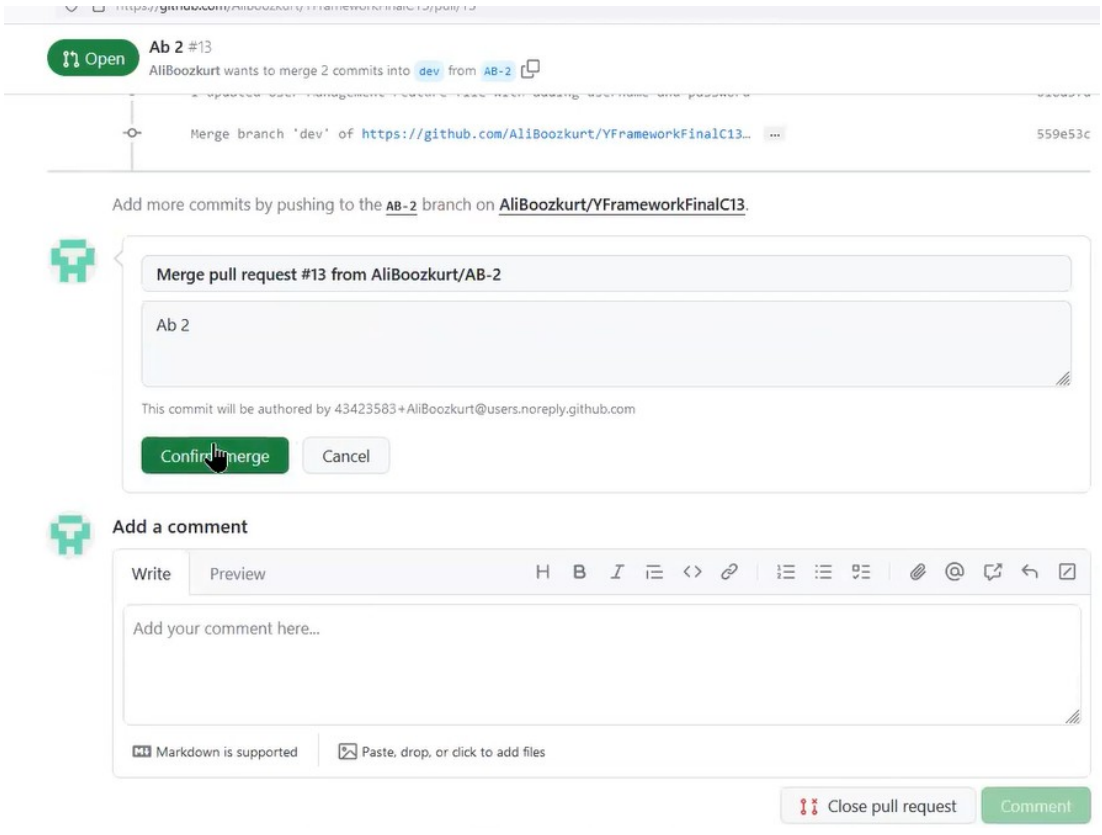
Labels

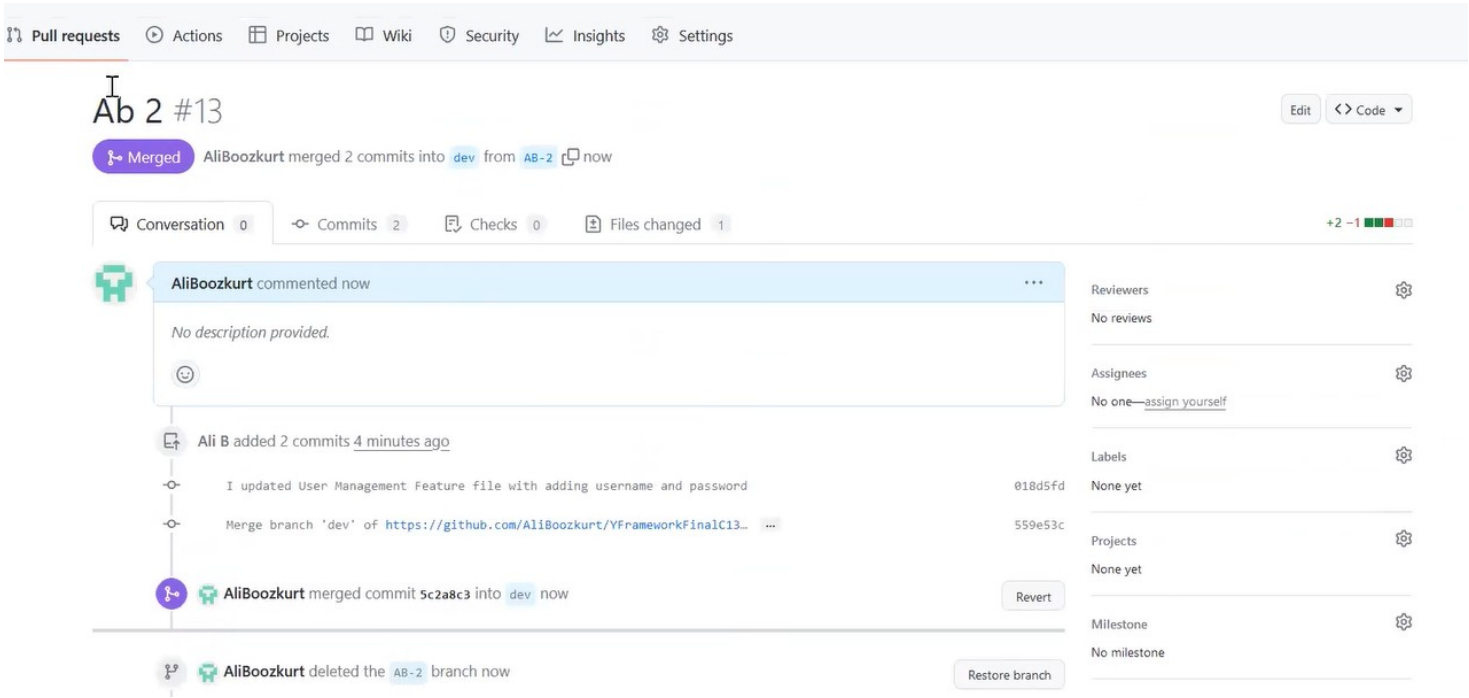
Markdown is supported

Paste, drop, or click to add files

Create pull request


Open
Ab 2 #13
AliBoozkurt wants to merge 2 commits into dev from AB-2
Merge branch 'dev' of https://github.com/AliBoozkurt/YFrameworkFinalC13...
559e53c
Add more commits by pushing to the AB-2 branch on AliBoozkurt/YFrameworkFinalC13.
Checking for ability to merge automatically...
Hang in there while we check the branch's status.
Merge pull request
You can also open this in GitHub Desktop or view command line instructions.
Add a comment
Write
Preview
Add your comment here...
Please fill out this field.
Markdown is supported
Paste, drop, or click to add files
Close pull request
Comment

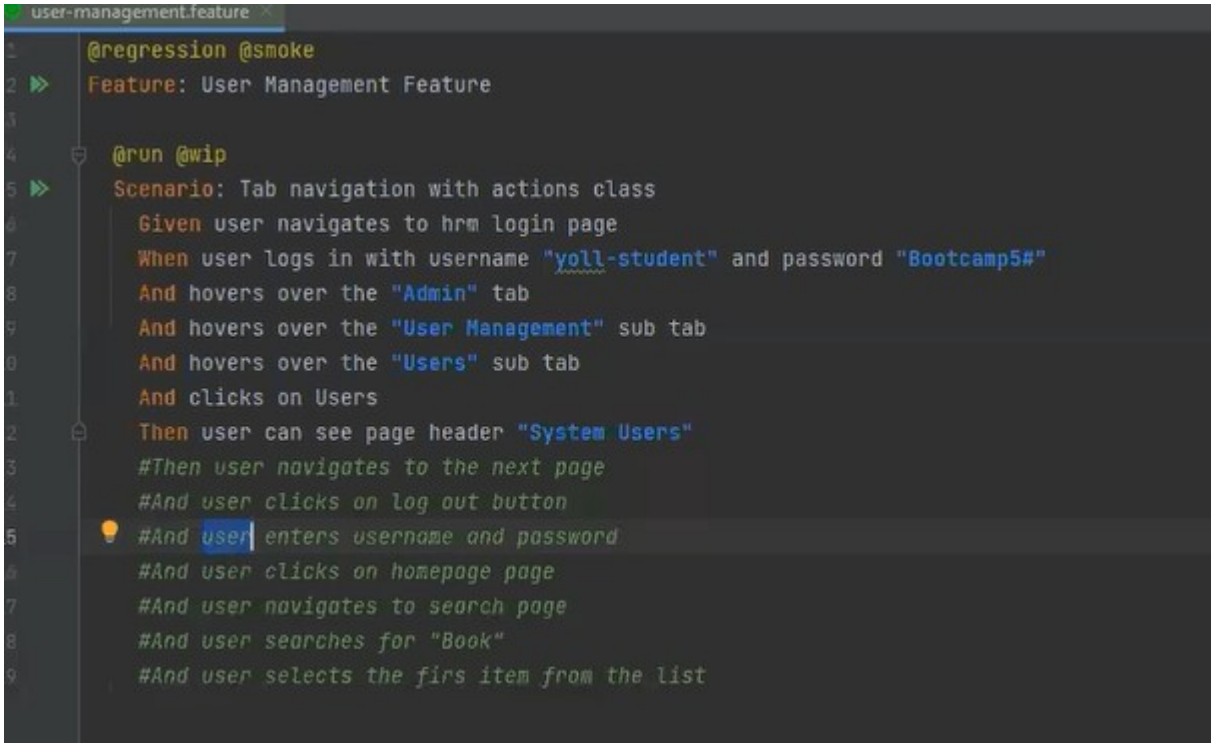

Open
Ab 2 #13
AliBoozkurt wants to merge 2 commits into dev from AB-2
Merge branch 'dev' of https://github.com/AliBoozkurt/YFrameworkFinalC13...
559e53c
Add more commits by pushing to the AB-2 branch on AliBoozkurt/YFrameworkFinalC13.
Merge pull request #13 from AliBoozkurt/AB-2
Ab 2
This commit will be authored by 43423583+AliBoozkurt@users.noreply.github.com
Confirm merge
Cancel
Add a comment
Write
Preview
Add your comment here...
Markdown is supported
Paste, drop, or click to add files
Close pull request
Comment



Now it is merged.

Go to IntelliJ

Checkout dev


user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And user enters username and password
#And user clicks on homepage page
#And user navigates to search page
#And user searches for "Book"
#And user selects the firs item from the list

Use : git pull

we need to update dev by using : PULL. Otherwise, we can see the change localy on the two branches: AB-1 and AB-2

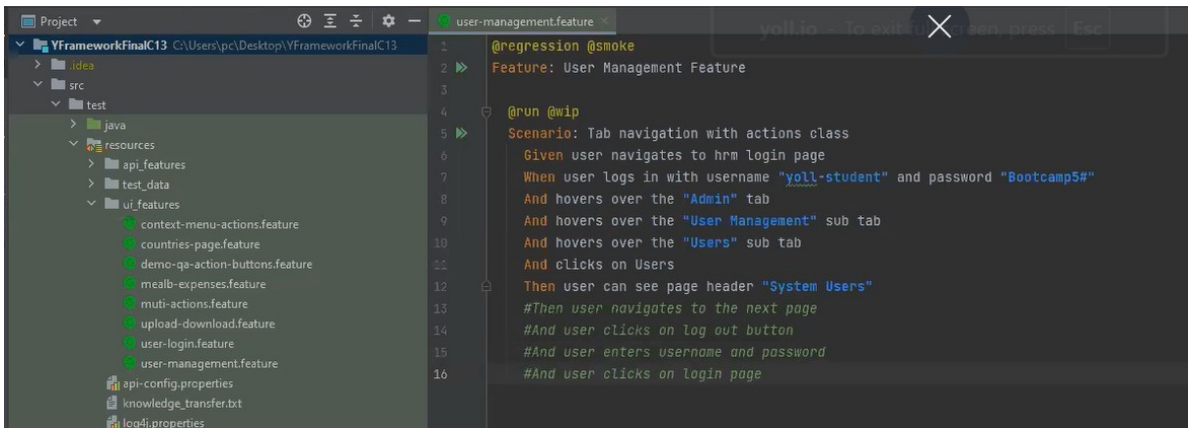
Execute : git pull

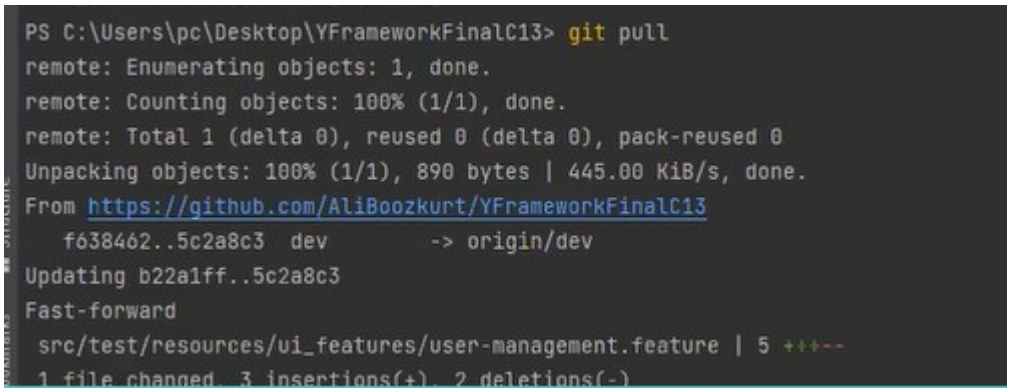
- `git pull` brings changes from a **remote repository** (e.g., GitHub) **into your local branch**.

Dev :

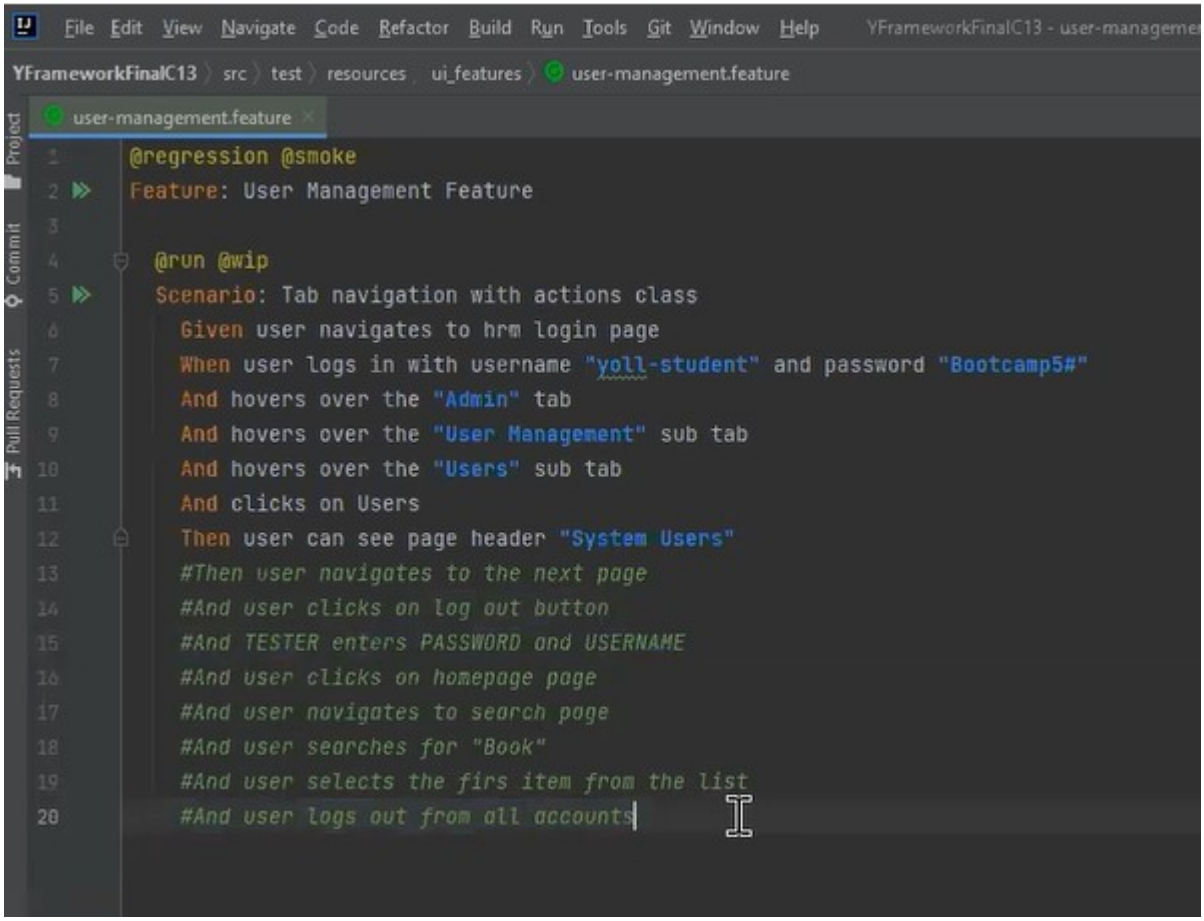
Checkout dev first

You have to update dev localy by using pull command because the changes happened only on dev Remote on GitHub




PS C:\Users\pc\Desktop\YFrameworkFinalC13> git pull
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (1/1), 890 bytes | 445.00 KiB/s, done.
From https://github.com/AliBoozkurt/YFrameworkFinalC13
f638462..5c2a8c3 dev -> origin/dev
Updating b22a1ff..5c2a8c3
Fast-forward
src/test/resources/ui_features/user-management.feature | 5 +++--
1 file changed, 3 insertions(+), 2 deletions(-)

The final result on IntelliJ:


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
YFrameworkFinalC13 - user-managemer
YFrameworkFinalC13 src test resources ui_features user-management.feature
user-management.feature
@regression @smoke
Feature: User Management Feature
@run @wip
Scenario: Tab navigation with actions class
Given user navigates to hrm login page
When user logs in with username "yoll-student" and password "Bootcamp5#"
And hovers over the "Admin" tab
And hovers over the "User Management" sub tab
And hovers over the "Users" sub tab
And clicks on Users
Then user can see page header "System Users"
#Then user navigates to the next page
#And user clicks on log out button
#And TESTER enters PASSWORD and USERNAME
#And user clicks on homepage page
#And user navigates to search page
#And user searches for "Book"
#And user selects the firs item from the list
#And user logs out from all accounts

GIT BEST PRACTICES

- **Pull regularly,** and especially before commiting.
- Don't commit to Master Branch. (That is what production is based on) You will not need to change anything on Master on real project. You will be involved with Dev/Test branches.
- Commit **related changes** together
- Commit **often**
- Do not commit half-done work (stash instead if needed)
- **Test code** before you commit
- Write **informative** commit messages

The longer you wait the more conflict you have.

It is easier to fix two lines than ten and twenty.